

1-1-1979

Guidelines for the in-house development and implementation of a systems methodology.

Warren A. Norris

Follow this and additional works at: <http://preserve.lehigh.edu/etd>



Part of the [Industrial Engineering Commons](#)

Recommended Citation

Norris, Warren A., "Guidelines for the in-house development and implementation of a systems methodology." (1979). *Theses and Dissertations*. Paper 1898.

This Thesis is brought to you for free and open access by Lehigh Preserve. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of Lehigh Preserve. For more information, please contact preserve@lehigh.edu.

GUIDELINES FOR THE IN-HOUSE
DEVELOPMENT AND IMPLEMENTATION
OF A SYSTEMS METHODOLOGY

by

Warren A. Norris, Jr.

A Thesis

Presented to the Graduate Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Industrial Engineering

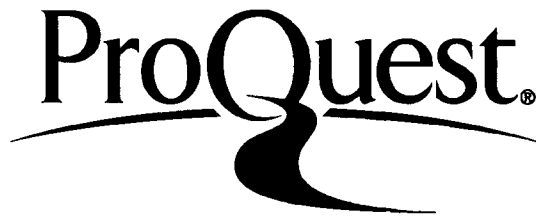
ProQuest Number: EP76170

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



ProQuest EP76170

Published by ProQuest LLC (2015). Copyright of the Dissertation is held by the Author.

All rights reserved.

This work is protected against unauthorized copying under Title 17, United States Code
Microform Edition © ProQuest LLC.

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

This thesis is accepted and approved in partial
fulfillment of the requirements for the degree of
Master of Science.

4/27/49

(date)

Professor in Charge

Chairman of Department

ACKNOWLEDGEMENTS

There are many individuals whose names should appear in this page for providing assistance to me in the preparation of the thesis. For that aid I have been very thankful and have told them so. However, there are two individuals and an entity whom I must gratefully acknowledge for contributing not only the majority of the technical assistance but the support needed to bring the thesis to fruition. This gratitude, not adequately expressed, is directed to:

Bethlehem Steel Corporation, without whose support through the Educational Assistance Program, the graduate course of study might never have been undertaken,

Dr. Ben Wechsler, without whose challenge and inspiration I might not have persevered to complete the course of study, and

My wife, Nancy, without whose encouragement, it would never have been attempted in the first place.

TABLE OF CONTENTS

	<u>Page</u>
ACKNOWLEDGEMENTS	iii
TABLE OF CONTENTS	iv
ABSTRACT	1
 Chapter One - INTRODUCTION	
1.1 Background	5
1.2 The Problem	13
1.3 The Thesis Objective	14
 Chapter Two - THE APPROACH USED IN DEVELOPING GUIDELINES	
2.1 General	15
2.2 Literature Review	16
2.3 A Mail Survey	21
2.4 A Case Study	23
2.4.1 Company Background Information	24
2.4.2 The Methodology Development Project	27
2.4.3 Chronology of Events	28
2.5 Conclusions	44
 Chapter Three - GUIDELINES FOR DEVELOPING SPECIFICATIONS	
3.1 Creating the Necessary Environment	45
3.1.1 Need Recognition	46
3.1.2 Data Processing Management Support	48
3.1.3 The Data Processing Organization	50
3.1.4 The User Environment	53

Table of Contents (cont'd)

	<u>Page</u>
3.1.5 Basic Concepts	55
3.1.5.1 User-oriented project management	56
3.1.5.2 A return on investment concept	58
3.1.5.3 Support of organizational decision-making	60
3.1.5.4 Higher quality initial design	60
3.1.5.5 "Creeping Commitment"	63
3.2 Specification Development Procedures	64
3.2.1 Consider the effort as a "project"	65
3.2.2 Define the Problem, Objectives and Scope	65
3.2.3 Assign an individual or a team	67
3.2.4 Apply a methodology to your own project	69
3.2.5 Develop a project plan	70
3.2.6 Review the existing literature	71
3.2.7 Visit other firms	72
3.2.8 Consult the internal staff	73
3.2.9 Consider the Components of your Methodology Manual	74
3.2.10 Write a Specifications Document	94
 Chapter Four - MAKE VERSUS BUY CONSIDERATIONS	
4.1 Introduction	96
4.2 Identify Existing Components	97
4.3 Review Commercial Packages	98
4.4 Establish Interface Requirements	102
4.5 Estimate Manpower and Duration Requirements	103
4.6 Perform a Cost/Benefit Analysis	104

Table of Contents (cont'd)

	<u>Page</u>
4.7 Select An Alternative	107
4.8 Submit a Proposal	109
 Chapter Five - GUIDELINES FOR DEVELOPING YOUR MANUAL	
5.1 Introduction	111
5.2 Provide for Project Control	111
5.3 Plan For In-House Development	112
5.4 Establish a Review and Approval Procedure	115
5.5 Establish Documentation Standards for Your Manual	119
5.5.1 Organization, Format, and Content	120
5.5.2 Pagination	122
5.5.3 Glossary	124
5.5.4 Standard Documents	124
5.6 Logistics	125
5.6.1 Writing	126
5.6.2 Typing	127
5.6.3 Proofreading	127
5.6.4 Reproduction and Preparation	127
5.6.5 Distribution and Mailing	128
5.6.6 Editing	128
5.6.7 Updating	129
 Chapter Six - IMPLEMENTING YOUR METHODOLOGY	
6.1 Phased Implementation	131
6.2 Provide for Education of All Staff Members	133
6.3 Provide for Follow-up	135

Table of Contents (cont'd)

	<u>Page</u>
6.4 Provide for Maintenance of the Methodology Manual	137
6.5 Establish a Post Implementation Review	137
Chapter Seven - CONCLUSIONS AND AREAS FOR FURTHER STUDY	
7.1 Conclusions	140
7.2 Areas for Further Study	143
▽	
BIBLIOGRAPHY	146
APPENDIX	
Exhibit A	149
Exhibit B	153
VITA	158

ABSTRACT

DEVELOPING AND IMPLEMENTING AN IN-HOUSE
SYSTEMS METHODOLOGY

This thesis presents practical guidelines for the in-house development and implementation of a systems methodology manual. As computer applications have become more complex, and as data processing, as a business function, has matured, many have observed the failure of a large number of system development efforts. Such disappointments have illustrated the need for a formalized application development strategy. In recognition of these problems, many authors, consultants, and educators have recommended various project life cycles, task lists, tools for analysis and design and coding methodologies. Some have assembled these aids and developed commercial methodologies which have been marketed much like computer software packages.

Given that an organization would like to adopt a formal development strategy, a set of guidelines on the subject of how to go about obtaining and/or developing such a methodology in-house does not currently exist.

The objective of the thesis is to provide Information Systems Departments with an alternative other than purchasing a commercial methodology or the hit or miss development of one's own methodology.

The guidelines were based largely on a case study, in which the in-house development of a Methodology Manual by a large corporation was observed. A mail survey and search of the literature were other resources employed in the preparation of the guidelines.

Prior to developing one's manual, the guidelines advocate organizing the in-house effort as a project. In addition to traditional project tasks, the guidelines also insist upon the development of a specifications document. Such a document defines the components and features which are to be included in the in-house methodology. The components of one's methodology manual should include at least the following:

- Project Acceptance and Approval Procedures
- Creeping Commitment Policy
- User Responsibility Policy
- Development Life Cycle including Phase Tasks
- Project Milestones and Milestone Documents
- System Assurance Reviews

Compiling a specifications document places the project team in a better position to review commercial methodologies and/or to write their own.

Although concerned primarily with the in-house development of a methodology manual, the guidelines also propose procedures for the objective comparison of the purchase of a commercial package as opposed to the development of the manual in-house. Procedures for this analysis include the following:

- Identify existing components (using the specifications document)
- Review commercial packages
- Establish interface requirements
- Estimate manpower and duration
- Perform a cost/benefit analysis
- Select an alternative
- Submit a proposal

If in-house development is pursued, various development steps are recommended such as:

- Assign quality, design and project control responsibilities
- Plan for in-house development
- Establish a review and approval procedure
- Establish documentation standards (including organization of the manual, its format and pagination rules)
- Perform operating details such as writing, typing, editing, proofreading, correction, editing, updating reproduction, distribution and mailing.

Although other implementation approaches are discussed, the guidelines recommend that one's manual be implemented chapter by chapter, with each phase of the life cycle assigned its own chapter. The manual would then be implemented as a system might progress through the development cycle. Such a sequence would also provide built-in milestones for project control.

Chapter One

INTRODUCTION

1.1 Background

Historically, approaches to the development of information systems within most data processing organizations have been characterized by inconsistency. There were either no standardized system development methods or, if an approach or development strategy was employed on a project, it was usually imposed by the individual project leader for that development effort alone. In the more recent past, as computer applications have become more complex, and as data processing organizations have grown, there has been increasing evidence of the necessity for controlling data processing resources and for developing good systems in a reasonable time and at a reasonable cost. The need for guidelines for the successful development of good systems has been illustrated over the years by the failure of many development efforts, some of which were more spectacular than others, as shown by the following examples:

- B.S. Nelson, in his article, Who Needs Methodology?, mentions a "thirty-one million dollar lawsuit in the Courts of Switzerland against a company which failed to complete a teleprocessing system for a Swiss bank within the promised time period." The company supposedly had invested over three hundred man-years in the development of the necessary software without success (1).
- E. Yourdon, in Design of On-line Computer Systems, discussed General Electric Company's initial development effort for the MEDINET hospital information system, a classic case of lack of success in system development. The initial project was terminated after two years of intensive work failed to produce the contracted product (2).

Based on the author's experience, most data processing practitioners have encountered many similar war stories. While both Nelson and Yourdon indicate that there were many reasons why these development efforts

failed, a strong case exists that adherence to a development methodology could have improved the respective chances for project success. This point has been emphasized by P. W. Metzger, who suggests that poor planning is largely responsible for schedule and cost overruns on (system) development projects (3).

There are, however, additional benefits other than prevention of cost and schedule overruns which can be attributed to the effective utilization of a formalized application development methodology. These would include the following:

- The development of systems which are technically and operationally adequate, as claimed by J. D. Toellner, in User Involvement and Enforceability - What, When, and How. He considers appropriate user involvement in the development process of paramount importance to project success. He suggests that "a clear definition of the life cycle (Ed. - a major component of a formal methodology) will cause the needed user involvement activity to happen in the right sequence." (4)

- The ability of top level management to exercise "reasonable control over the ever-volatile, high technology data processing function" in order to prevent the placing of marginal or unprofitable applications on the computer, a practice decried by many consultants and observers (5). Adherence to a methodology which requires a thorough cost/benefit analysis, together with a project approval mechanism, can shift the onus of project selection and/or approval to top level management in order to provide that control.

These examples and references only scratch the surface of the arguments presented by available literature advocating adherence to a formalized system/project development methodology. Each of a number of software houses, consultants, authors, and educators has proposed, written or recommended a separate version of a development project life cycle together with an associated methodology or set of application development guidelines. Some of these are offered freely in trade magazines and through vendor sponsored user organiza-

tions; others, considered proprietary information, are revealed only at a considerable price. The value of the respective methodologies will not be debated further in this thesis. On the contrary, and in spite of the variety of methodologies which have been proposed, the author concurs with those system developers who believe that data processing systems are more likely to be successful if good development practices are consistently followed within a disciplined environment. Such an environment can best be provided by consistent adherence to a formalized application development strategy.

Most books and articles on the subject of systems development tell us why we need a "Methodology" and then proceed to describe their Project Life Cycle, tasks, responsibilities, tools for analysis, programming aids, and documentation forms. As a result, the articles serve to help data processing managers recognize the necessity for a development strategy that will provide a disciplined environment for the successful development of good systems. This is quite an accomplishment in itself. The literature also helps

to formulate ideas as to what some of the components of the Methodology should be. Nevertheless, these texts, although valuable, do not go far enough. They do not provide guidelines for those who may choose to:

- Develop and implement their own methodology, or
- Purchase a package, modify it to fit their needs, and then implement it.

System developers, thus enlightened by the existing literature, have in many cases attempted to install some of the recommended methodology components within their own organizations in a piecemeal fashion, a practice which results in an extended implementation period. A survey relative to application development methodologies which was mailed to the Manager/Director of Management Information Systems of several companies, both large and small, some of national orientation, some local, supports this observation. Of the forty survey respondents, fifteen did not adhere to a development methodology at all. Of the twenty-five that did, the average development and implementation period reported was approximately three years. Only two of the twenty-five had implemented all eighteen of the

major methodology components of interest which had been listed in the questionnaire. Those who purchased a packaged methodology were required to devote additional time and expense to modify the package to suit their particular organization, as follows:

<u>Company</u>	<u>Effort</u>	<u>Elapsed Time</u>
A	2 man-years	1 year
B	4+ man-years	2-1/2 years
C	"considerable"	5 years
D	"considerable"	"more than 1 year"
E	2 man-years	1 year

In addition to a piecemeal and lengthy implementation and a large investment of man-hours, attempting to develop and install a formal methodology--in-house, without the benefit of any guidelines, could result in an unhappy experience and/or wasted time. For example, the survey revealed that twenty firms had developed and installed their methodologies without the benefit of guidelines or outside (vendor/consultant) assistance. Of this group, five claimed that their methodologies were not entirely successful. At the same time, one of

five who purchased an application development package was not convinced his organization would proceed in the same fashion had they an opportunity to start over again (see Company D above).

It would appear that a methodology developer, when faced with the task of developing and implementing a System/Project Development Methodology for his own organization, has a need for answers to the following questions:

- Where does one begin?
- What should be the scope of our methodology?
- Which life cycle and which methodology are most suitable for my organization?
- How shall we select a particular methodology for our use?
- What alternatives are available, such as purchase or in-house development?
- Which "package" should we purchase?
- Must purchased materials be modified for our own use?
- What are the requirements for writing our own methodology?

- How can we ensure that the staff will accept the methodology without compromising its purpose?
- What are the pitfalls of the various approaches?

Research of available literature for assistance in implementing a System/Project Development Methodology revealed a considerable amount of information on the subject of how to develop a data processing or management information system or on what the contents of a methodology should be. In fact, the plethora of information on these subjects is part of the problem. There appear to be so many different approaches to systems development, that one can easily become confused as to which is the most suitable. On the other hand, the search for answers to the questions posed above failed to reveal much in the way of guidance for the in-house developer.

1.2 The Problem

Given that an organization does not have, but recognizes the need for, a formal development strategy, a set of guidelines on the subject of how to go about obtaining and/or developing such a methodology does not currently exist. In many instances this results in

either the purchase of a methodology, which requires extensive modification, or the hit or miss development of one's own methodology.

1.3 The Thesis Objective

The objective of this thesis is to present practical guidelines for the development and implementation of a formalized applications development strategy. Insofar as possible, these guidelines will be based on the experiences of individuals and companies who have attempted such a task. Done in this fashion, it is expected that the guidelines will provide Information Systems Departments with a charted course which, when followed, will:

- Improve their planning for Methodology development,
- Help them move off dead center in the development effort, and
- Assist in the recognition and avoidance of time-consuming and costly pitfalls.

Chapter Two

THE APPROACH USED IN DEVELOPING GUIDELINES

2.1 General

In order to accomplish the thesis objective, i.e., the presentation of "practical guidelines for the development and implementation of a formalized applications development strategy," appropriate research was first necessary to accumulate the information base from which the guidelines would be constructed. The intention was to seek information that would provide a better understanding of:

- the potential situations and decisions faced by the in-house developer,
- the alternative solutions considered,
- the actual outcomes of decisions made,
- the specifications required for a development methodology, and
- the environment needed for a successful methodology development effort.

The proposed research should, therefore, uncover documentation of the recommendations of methodology

developers who have written about their experiences. This could be accomplished by reviewing the available literature and also by conducting a mail survey of companies and institutions utilizing Management Information Systems departments. In addition, the research should also develop some new insights. The latter could be gained by observing the actual construction of a Development Methodology Manual for a large manufacturing concern and by documenting the steps taken.

2.2 Literature Review

In the search for the documented experiences of others as to how to go about obtaining a formal development methodology, the following types of sources were reviewed, some examples of which have been referenced specifically in the thesis:

- Textbooks. Many were reviewed, all of which had something to offer the data processing practitioner with respect to the development of computer systems. Several furnished valuable input for developing specifications for various methodology components. None of those reviewed, however, provided the sought-after guidelines

for developing and implementing a Development Methodology in-house.

• Periodicals. These were magazines published primarily for individuals involved with management information systems as users, developers, operators, educators, or manufacturers of hardware, and included issues of Computer Decisions, Datamation, Data Management, Data Processor (IBM), EDP Analyzer, ICP Interface, Infosystems, Journal of Systems Management, IEEE Transactions on Software Engineering, among others. Of all the articles reviewed in these periodicals, only one was found which actually came close to the sought-after guidelines. It was written by Messrs. Frazier, Haugg and Thackery of Liggett and Myers, Inc. entitled, Developing a Project Management Package (Journal of Systems Management, December 1976), p. 39-40 (6). The article contained a very brief description of some of the steps taken by a particular organization to produce a formal methodology in-house. In the article approximately ten tasks were discerned which, although of value to the neophyte methodology developer, were not discussed in any detail nor were they presented in a

guideline format. Nevertheless, these points were valid concerns and will be expanded upon in the recommended guidelines to be presented in a subsequent chapter. As in the case of textbook sources, other articles found on the subjects of systems development, project management and control, or development methodologies presented a variety of advice, rules, and tools which should more appropriately be included in one's methodology specifications or within the body of the methodology itself.

- Promotional literature for commercial packages. Although there are many more vendors available from whom similar information can be obtained, only those for SDM (Systems Development Methodology) from Atlantic Software, Inc., Spectrum from J. Toellner and Associates, and Managing the Application Development Process (an audio course) from IBM were reviewed. The first two, although providing insights into the components of their respective methodologies, obviously did not provide guidelines for developing one's own methodology. The promotional literature for the third, an Independent Study Program (ISP) course offered by IBM for

approximately \$600, included the following modules among others on the list of course materials:

- "Components of the Application Development Process
- Implementing the Application Development Process
- Establishing Guidelines for Development"

These module titles indicated that the course might satisfy our needs, however, the price for the materials was too expensive to establish that as fact. In any case, the information was proprietary and could not be presented in this paper.

• Non-proprietary methodology manuals from other organizations. Copies (or representations of their contents) of the Development Methodology Manuals for the following organizations were obtained and reviewed:

- Frito-Lay, Inc.
- GTE Data Services, Inc.
- IBM - Application Development Cycle (Productivity Marketing and Requirements)
- Martin Marietta
- McDonnell Douglas (Design Review Process)
- Montgomery Ward
- Monsanto Chemical Co.

- Orange County, California
- Pennzoil Company

In the absence of guidelines for the development and implementation of a formalized application development package, the above manuals were by far the next best thing. One could readily synthesize some of the methodology development concerns or steps for the guidelines simply by viewing the finished product of someone else's efforts. The manuals were different in almost every conceivable way except for functional purpose and, in some respects, their components. Nevertheless, advantages and disadvantages of certain aspects of content and organization could be readily exposed by comparison.

- Consulting services reports. The Auerbach Information Management Series, The Diebold Research Program and Datapro Research Corporation reports provided considerable information relative to methodology principles, purposes, and component specifications similar to that found in textbook sources.

- Seminars and User Group Meetings. "Continuing Education" type seminars, such as those sponsored by the

American Management Association (AMA), various universities and consulting firms, in addition to IBM's DP Managers' Interest Series on the Application Development process, and meetings of users groups such as SHARE and GUIDE, should furnish insights into what others are doing or recommending in the area of methodology development. The course description for a seminar entitled, In-House Development of Data Processing Documentation and Procedures Manuals, offered by the Industrial Engineering Department of Lehigh University during the summer of 1978 indicated that the seminar would address the subject of this thesis. Lack of time and resources prevented sufficient opportunities to develop this area further.

2.3 A Mail Survey

In order to establish in an empirical fashion what other organizations had actually accomplished relative to methodology development, a questionnaire was composed (See Exhibit A in the Appendix) and mailed to eighty(80) manufacturing companies throughout the country. Some statistics developed from the survey re-

sponses were presented earlier in Chapter 1.0. There were forty questions directed to the Manager or Director of Management Information Systems at those locations. Eight were designed to establish some idea of the size of both the company and its data processing organization as well as the scope of its management information system involvement. The remainder of the questions dealt with System/Project Development Methodologies. Eighteen of the questions were designed to determine what components were included in the respective methodologies, having established earlier that the "data processing organization adhere(s) to a formal System Project Development Methodology."

Other questions developed whether the respective methodologies had been purchased, or developed in-house with or without the benefit of guidelines and with or without the assistance of a vendor or consultant. The questionnaire attempted to establish who in the organization was responsible for methodology development (or for modification of a purchased package if that were the case), how many personnel were involved and the extent of that involvement. It also attempted to secure

some idea of whether or not the methodology had been successful, had the support of top management, and was being maintained on a continuing basis. Finally, responders were asked to indicate, with the benefit of hindsight, whether or not they would choose the same approach had they the opportunity to do it all over again.

Fifty percent of those surveyed responded to the questionnaire. Two factors were believed to be responsible for the very satisfactory return rate. First, a pre-stamped, addressed envelope was provided for ease of responding. Second, most of the survey questions were designed to be answered with a simple yes, no, or check-mark response. A summary of responses can be found in the Appendix, Exhibit B.

2.4 A Case Study

The major factor in the development of the proposed guidelines was provided by the observation of an in-house Methodology Manual development effort undertaken by one of the country's large corporations. That project gave the author the opportunity to document the problems, decisions, activities, and events connected

with it. As a result, the considerable amount of material garnered in this fashion plus the benefit of hindsight provided additional sources of information upon which to base the recommended guidelines.

2.4.1 Company Background Information

The subject corporation employed over 90,000 persons at the time of the project and had an annual data processing budget in excess of \$50 million. There were nine(9) geographically separate manufacturing facilities in addition to the corporate offices, each with a quasi-autonomous data processing organization reporting to corporate data processing management for policy direction. Because of this and other logical links together with the physical links between the processors at each of the locations, the nature of processing fits the definition of distributed processing. In other respects, such as departmental organization, size of staff, large central processors, and the absence of non-Data Processing programmers and computer operators, the nature of processing at the respective sites appeared to be highly centralized. Each of the ten locations had its own, separate Systems and Programming

Department. Corporately, there were a total of more than 250 Systems and Programming personnel available for the development of new applications and the maintenance of existing systems.

Prior to the Methodology Manual development project, neither the development of new applications nor maintenance of old systems had been performed under the aegis of a uniform, formalized application development methodology. There was, however, a standards function, assigned to the corporate data processing headquarters, which was primarily responsible for establishing documentation standards for program, computer operations, and user documentation. It was also responsible for standards in the area of naming conventions for describing programs, "jobs", hardware devices, and Job Control Language, etc. As it turned out later, many of these standards were incorporated into the documentation required or referenced by the new Methodology Manual.

In addition to the Standards activity, Long Range Planning procedures were in force for the data processing function. Because of the absence of a uniform

development approach, application development efforts included or planned for inclusion in the corporation's Plan of Projects did not identify with or relate to a commonly interpreted Project Life Cycle. The lack of uniformity, coupled with the absence of appropriate mechanisms for measuring progress against "The Plan," such as review and approval procedures, made management of the planning process difficult, if not ineffective.

The Company had also purchased project control software to assist the Systems and Programming departments. The operation of this system met with little success, and ultimately was used as no more than a timekeeping system for accumulating statistics, such as man-hours expended on system development projects. Such a system failed as a project control tool because the following had not been promulgated throughout the corporation:

- a uniform, phased project life cycle
- a standard, fully-defined phase activity list
- appropriate guidelines for systems development

There were also serious differences between departments and even within departments in the approaches to and

techniques used for designing system details, programs, and data bases. Internal communications were such that Data Base Administrators had insufficient opportunities to impact data base design. As a consequence of these and other conditions, project manhour and duration estimates were being missed by as much as 600%, systems were being developed but not implemented, and unjustifiable systems were being built. It became obvious to data processing management and, unfortunately, also to the user community, that changes had to be made throughout the corporation in the way systems were being developed.

2.4.2 The Methodology Development Project

Having recognized the not too subtle evidence of need for some sort of framework for systems development, the corporate management of the data processing function directed that a member of the corporate data processing manager's staff be assigned, full time, with the responsibility for recommending a solution to the above problems. It is interesting to note that the individual assigned was not a Manager or Supervisor of the Systems and Programming Department or anyone cur-

rently involved in the development of data processing systems. Nevertheless, he had many years of experience as a Systems Analyst and had been responsible for designing and implementing systems for plant manufacturing, home office, and corporate-wide applications. He had a working relationship with members of the data processing organizations at the remote locations as well as at the corporate offices and was well-informed as to application development efforts currently in progress. One reason for his selection may have been that he had also been the lone vocal advocate of a "more organized approach to systems development, one that was consistently interpreted throughout the organization and that paid attention to 'good' development practices." It was at this stage in September, 1976 that the company seriously started the effort which culminated in the installation of a System/Project Development Methodology in January, 1978.

2.4.3 Chronology of Events

The following chronology illustrates some of the more important aspects of the case study.

During September, 1976, a search and review of available literature on the subject of development methodologies and their creation was conducted. The search uncovered a wide variety of methodologies, but not much on their creation. In other words, the company reached the same conclusion with respect to the applicability of textbooks and periodicals as that claimed in a previous section (Section 2.2) of this thesis.

In October, 1976, a "white paper" was presented by the developer which recommended the adoption of a standard, corporate development methodology. The paper contained a statement of the problem, a statement of objectives and the rudiments of a methodology. The latter included a five-phase Project Life Cycle with descriptions of the related phase tasks. The number of phases was the sole constraint imposed upon the white paper's author by data processing management. This restriction caused some problems when it was time to implement the methodology; however, they were ultimately overcome. The first draft of the paper was circulated to the various data processing managers for review

and comment in order to seek a semblance of a consensus as well as additional input. This was believed to be necessary to combat the not-made-here (NMH) syndrome. The hope for a consensus proved to be ephemeral because of the variety of positions taken by the respective managers, ranging from active support to restrained rebellion. Nevertheless, all of their concerns and comments were addressed and tested against the statement of objectives and for consistency with the balance of the mini-methodology. Responses were published explaining the reasons for acceptance, rejection, or modification of each proposed change.

The resultant draft was submitted to the company's Data Processing Steering Group for review and approval. After approval by this body, the draft was reproduced and distributed to the various locations, together with instructions directing that, henceforth, new systems should be developed within the life cycle framework described therein. This document proved to be the outline for the company's System/Project Development Methodology. Because the mini-methodology lacked instructive guidelines, documentation, and other desirable

methodology components, the corporation took steps to obtain the desired product.

In December, 1976, a "Make or Buy" Study was undertaken. The writer of the preliminary version and a representative of the Home Office Systems and Programming Department were assigned first to review two commercial methodology packages, SDM (Systems Development Methodology) marketed by Atlantic Software, Inc. and PRIDE, marketed by M. Bryce and Associates. This task proved "easier said than done." Because of the proprietary nature of the packages, potential users were apparently expected to be able to make an intelligent decision without being able to view the contents of the respective methodologies. The Corporation, however, insisted upon the right to make a visual inspection of the methodology package before purchase. In the case of Atlantic Software, a non-disclosure agreement had to be signed by the evaluation team and a company Purchasing Agent before any SDM material could be reviewed. The evaluation of SDM addressed the following questions:

- Are the Corporation's planned methodology components included within SDM?

- Are the Methodology components offered by the package already being used in some form by the corporation? If one was not, was it already contemplated? In this case, a judgement was to have been made with respect to its advantages, disadvantages and its need for inclusion in the Corporation's System/Project Development Methodology.
- Will SDM give the control necessary to manage data processing projects and the corporation's data processing resources? If not, what additions or modifications are necessary?
- Are the benefits to be derived from using SDM commensurate with the costs for licensing the system?
- Will the use of SDM and its attendant vendor-supplied education program significantly reduce the time and/or expense to install a complete project management system?
- If the acquisition of SDM does not appear to be desirable, what alternatives should be considered?

- If acquisition of SDM does appear to be desirable, how should the Corporation implement the system?

The evaluation team concluded that the basic SDM offered little more than what was already being used by the Company. An expanded version of SDM, called SDM/70, with a larger price tag than basic SDM, was not yet available for proper review. Time had become a factor; so, further consideration of SDM was rejected.

The evaluators then visited a large multi-division corporation that had been using the PRIDE Methodology package for more than a year. This visit incurred needless expense, because preliminary communication between the two companies had not considered the proprietary nature of the package. As PRIDE users, they too, had signed a non-disclosure agreement. As a result, the evaluation team could not examine the contents of the package as it had been purchased from M. Bryce & Associates. However, some value was gleaned from the knowledge that this particular PRIDE user had expanded upon the purchased Methodology by writing additional development guidelines for project leaders, analysts, and programmers. The PRIDE users had also

eliminated one or two PRIDE life cycle phases dealing with evaluations or feasibility studies. The team also learned that their hosts had invested the efforts of a managerial level, multi-degreed person full time and three data processing management personnel part time in order to revise and implement the purchased methodology to suit their company's needs. The same personnel had an on-going responsibility for the education and training of the Systems and Programming staff and for updating the Methodology as required.

The evaluation team's management had concluded prior to the trip that if the visit to the PRIDE user did not prove promising, then the Methodology team should begin to develop a plan for in-house development. It was only left to the evaluation team to make a recommendation. They obviously could not make any kind of purchase recommendation based on knowledge of the contents of either package, SDM/70 or PRIDE. However, based on the PRIDE user's experience, they did conclude that their corporation would probably have to invest just as much time and effort to revise and implement a purchased package as it would to develop a

methodology in-house by expanding upon the "white paper." Based on this assumption and the fact that the corporation would not have to spend its needed cash for a package's purchase price, the evaluation team made the recommendation to take the in-house development route.

During the last weeks of December, 1976, the evaluation team developed the initial plan for completing the System/Project Development Methodology in-house. The plan was built upon a task list, a sequence, and estimates for each task. Modified Gantt Charts were prepared. PERT type arrow-diagrams did not seem to be applicable because the team could only visualize the writing resource for producing the manual. This lack of foresight in the first plan may have delayed actual project completion 2 - 3 weeks. If all tasks had been known, a better plan could have been prepared, together with more realistic estimates. Consequently, the first estimates for the project's duration were grossly underestimated by 100%. It was not quite as bad on a man-hour basis.

In January of 1977, the following were proposed and approved:

- A list of components for inclusion in the Methodology Manual
- A preliminary Table of Contents
- A format or organization for the Manual including pagination and Section/paragraph numbering.

The writing of the guidelines for the "Feasibility" Phase of the System/Project Development Cycle was started together with guidelines for an additional phase called "Evaluation." The latter was deemed as necessary in order to insert the interfaces for administrative tasks relative to new requests for services. Problems with the 5-phase constraint were avoided by simply not calling "Evaluation" a life-cycle phase. Introduction and General Overview sections were also prepared.

In February, while these sections were being drafted, three potential problems surfaced. The first involved the necessity for providing for insertions of complete sections or sub-sections or the addition of pages to the manual at any stage of its development or use. This

problem was believed to be resolved by beginning all sub-sections and chapters on a new page. It was also decided to identify major sections and sub-sections within the pagination scheme, numbering pages consecutively only within sub-sections; for example, 10.20.01, 10.20.02, . . . 10.20.99, etc. The second problem involved delays experienced by the writer while locating and expediting typing services, proofreading typed materials, duplicating copies for review purposes, distributing copies, and arranging for review meetings. As a PERT diagram could have shown, the drafting and editing of the written materials were the critical activities. The problem, to reduce the project's critical path, was solved by assigning responsibility for the administrative tasks to another staff member. The third potential problem was related to the review and approval process for the written guidelines. Initially, the Corporate Data Processing Manager and his two Assistant Managers desired to review and discuss every section and sub-section of the written guidelines prior to giving their approval for releasing any part of the Manual. This approach called for scheduling review

meetings dependent upon the availability of all three. The latter was a rare occasion, which tended to delay the review process. In addition, there was much discussion on every point, with the writer having to justify each position taken. It became apparent that this procedure was taking too much of the managers' time and, if continued, would extend the manual's development beyond a reasonable date. This type of review was soon ordered to be discontinued and replaced by some other review process involving the remote locations, as well as the home office data processing organization.

In the latter part of February, the project was brought to a temporary halt by two unscheduled interruptions. The first was caused by a management-requested survey which required the services of the Methodology writer. The survey was related to the administrative steps taken and analyses performed on new requests for data processing services prior to project approval by the management steering group. Although related to the Evaluation section of the Methodology Manual, the survey provided no new information, and the Evaluation

section was left unchanged. The second interruption caused the Methodology writer to be assigned to the "system assurance" type task of critiquing or evaluating various "phase-end reports" prepared by the Systems and Programming departments for some current projects. This served an educational purpose in that live examples of both good and bad practices could be discussed with members of the systems development staff. Nevertheless, both interruptions caused delays in the development of the methodology manual.

The first drafts of the Introduction, Overview, Evaluation, and Feasibility Phase sections of the Manual were completed in March, 1977. At this point, a review of the project's action plan was made, resulting in some significant changes. In order to avoid repeating earlier delays and to satisfy management's request for a review process involving all data processing organizations, it was resolved that the Methodology Manual development and implementation would proceed by Section and by various stages for each Section. This process not only facilitated project control and sped the actual use of the Manual, but it freed the writer

to perform the critical task of writing the guidelines. The development stages or tasks, which were repeated for each section, were as follows:

- a) Writing of the guidelines by the Methodology writer. This included drafting of new documents and instructions for their use.
- b) Submission of the draft to another staff member for controlling subsequent administrative tasks.
- c) Drafting of a cover letter for distribution purposes.
- d) Typing of the draft into a finished format and the cover letter.
- e) Proofreading the draft and cover letter.
- f) Duplicating a standard number of copies of the draft and cover letter.
- g) Distributing copies of the draft and cover letter according to a standard distribution list.
- h) Review of the draft by those receiving copies (managers of the respective data processing organizations and the heads of the respective Systems and Programming departments). Those receiving copies were directed to suggest any

constructive changes they thought would be in order and to return their comments within two weeks to the Methodology writer.

- i) Review of the responses by the Methodology writer, incorporating in the draft those changes which were deemed appropriate. Responses which were not accepted by the writer were submitted to the department manager for resolution.
- j) Preparation of a report noting all changes which were accepted and made, as well as those which were not made, together with a reason for their rejection.
- k) Typing, proofreading, duplicating and distribution of the report.
- l) Preparation by the Methodology writer prior to final typing of a synopsis of the entire section noting any potentially controversial points. The synopsis was then submitted to the management steering group for their approval. It was usually discussed at their next regular meeting with the Methodology writer present and available for questions. Sections were generally

approved without alteration.

- m) Retyping and insertion of revised pages with the correct pagination. Updating the Table of Contents.
- n) Proofreading the final version.
- o) Duplication of the section in sufficient copies for access by the entire development staff.
- p) Distribution of final copies together with the updated Table of Contents and an appropriate cover letter.
- q) Implementation of the section's guidelines was to be effective upon release of the final version.

With this task sequence, a new section was in the process of being written while another was being reviewed, at the same time as an earlier section was being published. The remainder of the Methodology Manual was developed and installed according to these procedures. The only subsequent procedural change involved the use of word processing equipment and the storage of Methodology Manual information on magnetic tape cassettes for ease of revision.

It was also at this time that responsibility for project control was given to the writer's departmental manager. From this point, the development of the Methodology Manual moved steadily toward fruition, with the following exceptions:

- a) Several "experts" who were to have written various parts of the manual, relative to their areas of expertise, failed to do so in a timely manner. The Manual did not necessarily enjoy the same priority in different departments.
- b) Some data processing locations requested postponements in implementing the Manual's requirements. Resolution of these issues also delayed the entire process. In cases such as these, the Corporate Data Processing Management was adamant in their resolve to implement a common development approach throughout the corporation.
- c) Although included in the Manual's documentation section, copies of new forms were not always available for use by the development staff at the time of a section's implementation. Delays in obtaining the forms in quantity from the printer

seemed to cause some loss of momentum and/or credibility with respect to acceptance of the guidelines.

Adhering to the above schedule, the writer and those involved published a major section each month until the System/Project Development Methodology Manual was finished in January, 1978. In the project's sixteen months duration, the writer had devoted 65% of his available working time, or 1580 manhours, on tasks related to the Methodology. Others spent a total of 320 manhours writing various sections or subsections of the Manual.

2.5 Conclusions

The completed research has provided sufficient background and information from which the proposed guidelines can be written. The case study has provided the bulk of the information used whereas textbooks have supplied the least. As stated earlier, Development Methodology Manuals prepared by other organizations also provided many good ideas.

Chapter Three

GUIDELINES FOR DEVELOPING SPECIFICATIONS

3.1 Creating the Necessary Environment

The fact that a methodology is in place does not automatically guarantee that systems will be successfully built. This was substantiated by the mail survey. 20% of the respondents who stated that they had a development methodology also claimed that their methodologies were something less than "successful." Unfortunately, the survey did not solicit responses as to the criteria employed for judging the success or failure of a methodology. What conditions, then, should pre-exist in a company that would improve its chances for installing a satisfactory methodology--one that works? Based upon the case study, each of the following factors could have significant impact upon the environment surrounding the methodology development effort:

- Need recognition,
- Data Processing Management support,
- the organization of the Data Processing Department,
- the user environment, and
- the identification of sound business concepts.

3.1.1 Need Recognition

There are many combinations of circumstances which could occur that might cause a manager to label his methodology as unsuccessful. The following are only a few of the "symptoms" which could be blamed on failure to employ an acceptable, formal development strategy:

- Late projects
- Excessive manhour budget overruns
- Dissatisfied users
- Excessive system costs
- Excessive program maintenance
- Inadequate returns on data processing investment
- Failed (unimplemented/rejected) systems.

If any of these "symptoms" exist and are not of concern to anyone in the data processing organization, then examination of the development methodology being used will probably not follow. As was seen in the case study, the recognition of need for a change must occur to those who are in a position to do something about it. This is the first step in creating the desired environment for successful implementation of a development

methodology manual-need recognition. This does not mean that one should cause or wait for the above "symptoms" to happen. Many data processing managers have had the good fortune or good sense to try to profit by the mistakes and/or solutions of others before such symptoms began to appear in their own shops.

Obviously, if no one sees the need for such a tool, one will probably not be forthcoming,--unless yours is a divisional group and the corporate offices prevail upon you to install the home office package. Nevertheless, recognition of need for a development methodology may occur at many management levels, both within and external to the data processing organization. How high in the organization chart that this awareness goes may impact both the scope of the new methodology and the likelihood of its implementation. An example of this occurred in the case study. The widespread perception of the case company's requirements expanded the number of methodology components which had to be dealt with. Some of those added were Project Acceptance and Approval Procedures, Long Range Planning Interfaces, Project Management/Control Techniques, Program and Data

Base Design Methodologies, Change Control Procedures, Hardware/Software Selection and Post Implementation Reviews. The fact that both the Corporate Data Processing Manager and the Management Steering Group perceived the need for improvement in their approach to system development almost guaranteed that a Methodology Manual would be developed. As Nelson said in answering his own question, "Who needs Methodology? . . . Anyone whose present method of keeping track of things could be improved upon." (1)

3.1.2 Data Processing Management Support

A development methodology can be employed to some advantage by a project leader at the lowest level of management without the direction of his superior (providing contradictory procedures are not already in place). However, several methodology components will be missing because of the need for dealing with other members of the organization. Some of those mentioned earlier, such as Project Acceptance, Long-Range Planning Interfaces, Program and Data Base Design Methodologies, Change Control Procedures and Hardware/Software Selection, cannot readily be defined or performed by a

Project Team independently of the rest of the development staff. Each of the above-named methodology components as well as some of the individual tasks within the development cycle require the approval and active support of data processing management in order to be operable. This is particularly true if the development strategy is proposed from the ranks. It is also applicable if the development of the Methodology becomes the assigned responsibility of an individual or group of individuals below the level of Manager of Systems and/or Programming. According to the thesis questionnaire, 90% of the Methodologies which were constructed "in-house" were developed by the Manager of Systems and Programming or teams under his direction. In the case study, however, the project team was not under the direction of the Manager of Systems and Programming, but they did have the support of the Corporate Data Processing Manager and the Management Steering Group.

Where interfaces with company management external to the M.I.S. Department are involved, the Manager of Data Processing or Director of M.I.S. must support the

methodology venture, particularly if it is to be effective as it relates to the user community. The emphasis on management support goes beyond simply permitting the existence of formal development procedures and practices. The methodology must be researched, designed, written, edited, published, taught, promoted, implemented, and maintained (almost like a data processing system), all of which require expenditures of time and money. It takes considerable management support to authorize the large amounts of each required for developing a complete methodology, particularly one for a large corporation.

3.1.3 The Data Processing Organization

There are at least two aspects of the data processing organization which can affect the environment needed for successful development of your own methodology, namely, its type and its size. The type of data processing or M.I.S. organization will greatly influence one's ability to proliferate a standard philosophy and approach to system development. Obviously, a company with a single Systems and Programming shop will not be concerned with proliferation. On the other hand, where

a formal methodology is not in place, there is a good chance the management of a local Systems and Programming section can implement their version in their own shop with some degree of success (within the limitations discussed earlier). In larger corporations, a strong, centralized organization can hope to propagate a single standardized approach much more readily than can one that is decentralized or one that is centralized yet permits some measure of autonomy in its plants or divisions. The corporation in the case study had to overcome the opposition and desire for independence on the part of its divisional managers. Within other large corporations, small, autonomous systems and programming groups, serving specific user departments, can exist (for many reasons) and, depending on their relationship to and with the M.I.S. group, can have a positive or negative impact on the effectiveness of a corporate methodology. It is likely that they would tend to ignore any methodology but their own.

The size of an organization can affect the development of a methodology and its scope more so than the type of organization or management style. Small com-

panies may believe they do not have the required resources (time, money and expertise) to devote to the tasks necessary for implementing a methodology with all the components. They also may not notice the need, because the size and complexity of batch oriented applications run on smaller computer systems may be such that system development is more easily controlled. Staff size may be so small that the Manager of Data Processing serves as the Manager of Systems and Programming, Project Leader, Systems Analyst, and often the Programmer, all at the same time. In these case, there is no need perceived for approval procedures, technical reviews, cost/benefit analyses, documentation, and other controls. The most probable barrier to implementation of a development methodology in a small organization, therefore, is likely to be the lack of available time on the part of the Manager of Systems and Programming. Nevertheless, small companies cannot afford the inefficiencies tolerated by large ones. Problems of the small organization, therefore, should not be permitted to bar the installation of a formalized approach to system development.

3.1.4 The User Environment

The user is one of the more important ingredients in the bouillabaisse which will be your development methodology. Katch would go so far as to say,

"The best systems occur when the User conceives the solution because, in the end, it must be his system . . . through active user participation. Only when the ultimate user of the system makes the relevant decisions will it truly be his system." (7)

Both Katch and Toellner expand on that idea by maintaining that "appropriate user involvement" or participation must be defined by your methodology such that users will understand their responsibilities. Involvement must be "constructive and meaningful" in order to qualify as "appropriate" (7) (4).

If you seek the advice of the user community when defining the "User Responsibilities" component for your methodology, such a step may contribute indirectly to its continued success after it has been implemented. By an indirect contribution, it is meant that the interest and goodwill generated by personal contacts and consultations may improve the interpersonal relation-

ships of the respective areas and the data processing organization. If the data processing organization and the user community are in adversary positions, (for whatever reason, even the absence of a formal methodology), making the user aware of improved prospects for constructive, meaningful involvement in the development of future systems may establish an appropriate situation or environment for the successful development of your "System for System Development." In other words, whatever the current status of the user environment, user participation in the development of a user-oriented system development strategy will enhance the likelihood of its success. On the other hand, if the current relationships are poor, and your framework for the development of systems is constructed without user input or even awareness, the likelihood of near-term success will not be good.

Companies with established long-range planning procedures may require the data processing organization to interface with a formal body representing the user community, such as a Planning Council, Task Force or Steering Committee. If this is the case, and user management

participates in the development of the methodology, just as they participate in the development of the long-range plan, then you can expect their support and commitment of resources to this effort just as you can expect their "support and commitment of resources to the project in the long-range plan." (8) This also proved to be true in the case study. The user community, therefore, can make a very forceful, positive contribution to the success of your System/Project Development Methodology, providing that they become partners, in a significant way, in its creation.

3.1.5 Basic Concepts

In addition to the above "environmental" pre-conditions, there are several philosophies or concepts which, if not already identified within your organization, should be addressed before you proceed with the development of your methodology. They could have a significant impact on its content. Five of these can be expressed as:

- User-oriented project management,
- Return on investment policy,
- Support of organizational decision-making,

- Higher quality initial design, and
- Creeping commitment.

3.1.5.1. User-oriented project management is an extension of the ideas presented in the previous section. The thrust of this concept, however, is toward greater control of a system's design by its user(s) as opposed to the more traditional subterfuge described as "user participation". and "involvement" mentioned earlier, and which some writers describe as "pseudo-participation." This philosophy would become more visible (particularly to users) by the appointment of a user as Project Manager (a concept maligned by professional project managers).

One of the purposes for this approach is to ensure the development of systems whose quality is defined by user criteria as opposed to the data processing technician's inspired ideas. Another reason would be to minimize the OOPS (Ordinary Oversights Postpone Systems) factor (9) or at least some of the repercussions and wheelspinning when such oversights occur. Another would be to enhance

the likelihood that the completed system will be used, supported, and easily understood by those for whom it was designed (10).

Data processing organizations apparently have some distrust of this arrangement, particularly after going to the trouble of developing a "system for developing systems." The concern may be that the methodology, so carefully built for ensuring the above goals, may be subverted by the very same individuals whose interest it aims to protect. This would not be an area for concern if the condition for the appointment of a user Project Manager, together with his responsibilities and relationship to the development team, are fully described in your methodology. For those who still are not convinced, it is suggested that a condition of employment may be that the development methodology must be followed under the guidance of the data processing project leader or systems analyst. Nevertheless, user-oriented project management is suggested as one way to help users become jointly responsible with data process-

ing for the success of the systems they develop.

3.1.5.2. A return on investment concept may be employed within your company for the purposes of evaluating engineering, construction, or other projects which compete for the company's available capital resources. It may also become necessary, for example, for long-range planning purposes, to prioritize your system development projects. These efforts are also competing for limited resources-- your system developers, i.e., programmers and analysts. It is relatively easy to fit those projects which are considered "mandatory" into a priority queue, but not always so easy to rank projects considered "discretionary." A return on investment concept is therefore recommended for consideration as a ranking tool for prioritizing development efforts for the latter.

In addition, if one-time expenditures for equipment are high and/or one-time savings are also large, or if benefits will not be realized until a considerable time after a large development undertaking starts, a Discounted Cash Flow Analysis or

Present Value Analysis should be considered. Because of the nature of data processing systems, the likelihood is high for the consideration of systems which offer one-time savings brought about by benefits such as inventory reduction or cash-flow improvements, etc. In these instances traditional treatment by Discounted Cash Flow or Present Value Analyses may prove difficult. A solution may be provided by utilizing a similar, but compromise evaluation technique called a Profitability Index or a modification of it. This is simply the Present Value Earning divided by the amount of the investment. The resultant ratio is the Profitability Index (11). It is mentioned only as an example of a return on investment type of project ranking procedure which may be considered when developing your methodology. The use of these techniques and their input requirements must be described in your methodology if they are to be used in cost/benefit analyses and for presentation to management in phase-end and other milestone documents.

3.1.5.3. Support of organizational decision-making has not been a characteristic of conventional system development methodologies. This is not necessarily all bad. As stated in the Introduction, any methodology is better than none, and it can always be improved upon. Nevertheless, two methodologies will differ greatly if one encourages systems developers to focus primarily on paper flows and the other on decisions. Since there is considerable support for the latter in the literature to the extent that ". . . the true potential of computer-based systems will not be realized unless systems support decisions in the organization" (Auerbach, 1976) (10), it is suggested your methodology address both of these aspects of information systems design.

3.1.5.4. Higher quality initial design is a concept aimed at the need to restructure the thinking of those system developers, be they managers, designers, or programmers, who still believe a software project is not showing progress until "coding" has been started. This push to "code" on

the part of managers may come from their own "roots," i.e., their memory of times past when they had written software without the benefit of a flowchart. Others, possibly with some analytical and some programming responsibilities, may be fascinated with the technical aspects of computer systems, the power of the computer, and their own itch to interface with it. Whatever the reason, an organization operating under the premise that the quickest way to bring a system "on-line" is to start coding programs as soon as possible, may find itself experiencing longer development schedules. Methodologies developed in this sort of environment would tend to emphasize the generation of code early in the project life cycle.

The analysis of statistical data gathered over a long period during the systems development process by a large corporation enabled them to conclude that "over 50 percent of all development hours (had been) spent correcting bugs which result(ed) from faulty design." This claim was based partly on statistics relative to the times

and costs required to detect and resolve bugs encountered at various stages of development. They concluded that the development cost to detect and resolve software bugs and correct design mistakes increased almost geometrically as a system's development progressed. For example, the relative "development cost to detect and resolve a software bug after it (had) been placed into service (was determined to be) thirty times larger than the cost required to detect and resolve a bug during the early 'code reading' phase." (12)

Dr. Brooks, author of The Mythical Man-month, supports this view somewhat in that he states that he had for many years used a rule of thumb for scheduling software tasks in which $1/3$ of the scheduled time was allotted to planning and only $1/6$ to coding, the balance to testing. He claimed that the fraction devoted to planning was "larger than normal" (relative to coding) and that it was "barely enough to produce a detailed and solid specification, and not enough to include research or exploration of totally new techniques." (13)

Concluding that those projects which rush into the coding phase are those which ultimately have the longer total development times, one might tend to be guided toward developing a System/Project Development Methodology which places more emphasis on more thorough analysis and on "generating a higher quality initial design." Such a shift must promise a significant payback in terms of an overall reduction in project time, through reduced testing and fewer design changes (12).

3.1.5.5. "Creeping Commitment" is the process of providing management with multiple opportunities throughout a project's life cycle to review project progress, together with updated estimates of its cost and benefits in order that they may decide whether to proceed to the next step or not (14). Not all data processing organizations subscribe to this view. The writer is familiar with a particular data processing organization that at one time considered every request for its services a mandate and proceeded to develop a system right to the point of implementation with-

out a formal consideration of its good or weak points.

The literature suggests that this type of approach is a traditional one, that the "establishment of criteria for accepting the system has been done only after the fact, after time and money have been invested." It is also suggested that the criteria for evaluating a systems development project should be established prior to the end of each step with appropriate management reviews taken and authorizations to proceed given before the next step is started (15). Approvals to proceed would apply only until the next checkpoint(14). It is obvious that any policy or philosophy with respect to this aspect of system development within your organization will have an impact on your methodology and therefore should be clearly identified before proceeding further.

3.2 Specification Development Procedures

On the assumption that all of the environmental factors necessary for the successful development and

implementation of a System/Project Development Methodology are in place, and that the decision to proceed has been made, it is recommended that before you begin to write anything you first develop specifications for your Methodology Manual by performing the following steps, sequentially:

3.2.1 Consider the effort as a "project"

It will have a beginning and an end. It will have boundaries, and therefore will have a scope. It will have a problem or group of problems to solve or a current situation to change. It will have objectives. There will be alternative solutions with costs and benefits for each. There will be milestones and review points, plans and project control. It will have a specification phase, a production phase and finally, an implementation phase and possibly others which can be identified. If you do not organize the effort accordingly, development will be uncontrollable.

3.2.2 Define the Problem, Objectives and Scope

Prepare a statement which describes the problem or current situation, the objectives (of the methodology), and the scope of the project for the

benefit of management who must approve the effort and for the edification and guidance of those assigned the responsibility for the Methodology Manual:

- Identify and establish what the real motivation is for initiating the project.
- Describe the objectives of the methodology as it would deal with the stated problems.
- Identify the boundaries of both the project and its product, the Methodology Manual.

With respect to your project's scope, it may be necessary to establish that the expected solution of the problem is either an entire Project Management System which includes Long-Range Planning Procedures and Project Tracking, as well as the System/Project Development Methodology or just the latter. Next, determine to whom the methodology is to apply, e.g., Systems and Programming only, or will it apply to data processing management, users, and user steering committees. In addition, ascertain whether the methodology is to be applicable to more than one location or data processing organization, such as at a division or at a plant.

Potential Pitfall: Attempting to select components for inclusion in your methodology and setting priorities

for their development at this time may be slightly premature.

3.2.3 Assign an individual or a team to the effort.

As in any development project, your choice will have a major impact on the quality of the product and hence the success of the project. Expertise or experience in the following areas is required:

- Experience in the successful development of good systems (A sensitive individual who participated in a few fiascos may be a real plus.)
- Previous exposure to a successful methodology.
- Systems Analysis training
- Project Control training
- Program design, coding and testing
- Data Base design
- Writing

If these skills were embodied in one individual, so much the better; however, that is not likely to be the case. In the case study, the Methodology writer needed assistance in areas where he lacked experience, such as in Structured Design and data base design. The creditibility of your manual will be greatly enhanced if those who are to contribute to it are recognized as having expertise

in their respective subject areas.

The case study and the thesis survey also demonstrated that a considerable effort will be required, whether you develop your own Methodology Manual or you purchase a package from a vendor. Such an effort would conceivably be measured in man-years. Therefore, staff your project with full-time personnel and give them responsibility for the manual's creation. Then instill them, somehow, with a sense of urgency for completing the project.

Potential Pitfalls: Selecting personnel for assignment to the project based on their availability at the expense of skill level may be taking a risk. Educational background and experience may be inadequate for the task. Your Methodology will be weak in areas in which your writer is weak. However, either choice is a gamble. At one point in history, scientists claimed the earth was flat; their expertise went unchallenged until proven wrong.

Assignment of personnel part-time will postpone project completion. Sections of the manual will not be ready in a timely manner because of priority con-

flicts as were demonstrated in the case study. This could be particularly unfortunate when ideas and conclusions to be presented in the unfinished sections are prerequisites for the remaining sections.

The lack of urgency for completion will give an impression to the department as a whole as well as the team, that the Methodology Manual is not needed and is relatively unimportant.

3.2.4 Apply a methodology to your own project

Obviously, the author is suggesting that these guidelines serve as your methodology, one which recommends these major steps:

- Establish the proper environment,
- Work out the specifications for your manual,
- Consider the various alternatives such as purchase or in-house development,
- Produce the manual,
- Implement your manual, and make it function effectively.

As the author has claimed previously, any Methodology is better than none at all. So, if you choose to follow other steps, at least you considered an organized approach.

3.2.5 Develop a project plan . . .

at least for those tasks which have to be performed up to and including the selection of an alternative and submission of a proposal (see Sections 4.7 and 4.8). Further planning would have to be based upon assumptions relative to the selection of an alternative. Nevertheless, planning for implementation of the Manual, whether purchased or written, should enable you to give data processing management some idea of the project's overall duration.

Duration and manhour estimates will vary depending upon several factors, such as the amount of outside reading to be done by the project team, the number of components comprising the Methodology, and the number of reviewers and approvals required. In spite of these potential variances, it is hoped that the guidelines presented in this thesis will provide a more complete task list than that which was initially available to the project team in the case study. Awareness of the required tasks should enable you to develop a more accurate set of estimates.

It is also recommended that an activity arrow diagram, such as a PERT chart be used to establish the project's critical path. This technique should also identify those activities which can be performed concurrently by other members of the staff. The case study proved that planning, with a more complete task list and an arrow diagram showing predecessor-successor relationships, can shorten the development time for your project.

3.2.6 Review the existing literature

This will enable your project team to identify several methodology components and to support their guidelines by referencing published material. In addition to the periodicals referenced in Section 2.2 of the thesis, the following textbooks reviewed by the case study's project team still offer constructive advice for today, some better than others, on how to develop and design good systems. The reader can make his own choice as to which are best for him:

- Management Information Systems Handbook, W. Hartman et al. (16)
- Design of On-Line Computer Systems, E. Yourdon (2)

- Managing a Programming Project, P.W. Metzger (3)
- Systems Development Management Series, Auerbach Publishers (10)
- Structured Design, E. Yourdon, L. Constantine (17)
- Information Systems: Theory and Practice, Burch & Strater (18)
- Data Processing Project Management, T.R. Gildersleeve (19)
- 8-Stage Approach to Systems Development, GTE ta Services (20)

The most fruitful source of information will be the non-proprietary Development Methodology Manuals also referenced in Section 2.2 of the thesis. Try to obtain several similar manuals in order to view a wide variety of approaches.

A well-chosen team for your Development Methodology project would more than likely have read all these texts and as many more prior to getting their assignments. If this is not the case, your project is going to have to wait that much longer while they perform this review.

3.2.7 Visit other firms that have installed a System/Project Development Methodology in order to get a first-hand understanding of what is involved in developing, installing, and using this tool. Before

confirming your arrangements, however, you should attempt to avoid the same mistake that the company of the case study did: Don't visit a company that purchased a proprietary package. Legally, they cannot tell you anything about it. Besides, you might unwittingly compromise their integrity. A few suggestions; visit a company that:

- is fairly large in terms of either employees, sales volume, or capital investment. It is more likely to have a methodology that encompasses all types and sizes of applications than will a smaller one.

- has developed their own methodology. They might be proud enough to tell you about it.

- is nearby. Short trips can eliminate unnecessary expense, particularly if more than one day is required for review.

- has incorporated all the required methodology components as listed in Section 3.2.9.

3.2.8 Consult the internal staff, i.e., those who will have to labor within the framework you will provide. It was interesting to note that one of the criticisms suffered by the methodology developers of the

case study was that they "had not practiced what they preached." They did not make the effort to touch base with the "users" of the manual, the project leaders, systems analysts, and programmers, etc. who were to use the manual when it was completed. There were some legitimate reasons for that. For example, their group was already represented on the project team; there was insufficient time to interview everyone, and the practices and efficiency of the group were suspect. Nevertheless, the entire department never felt they had contributed to the development of the manual in spite of the review opportunities. It is recommended that you allow as many future methodology users as possible to participate in its creation by at least listening to their suggestions and opinions, whether they are used or not.

3.2.9 Consider the Components of your Methodology Manual

After your development team has reviewed the project's objectives and scope, researched relevant literature, visited other firms, and consulted the internal staff, they should be ready to identify the

components that will constitute your System/Project Development Methodology. What parts must we have and utilize in order to be able to claim that we have a bona fide Methodology? Toellner claims that the needed elements are:

- a clear project life cycle definition . . .;
- reasonable guidance to the staff on how to execute each of the tasks in the life cycle;
- management policies documented, issued, and enforced to give the needed authorities and to ensure that the process is used." (4)

Based on the author's research of the literature, the experience provided by the case study, and the results of the questionnaire mentioned earlier (Section 2.3), the above "elements" can certainly be described as the required components of a methodology and probably represent the smallest package which could be adopted that would be workable in any way.

For an attempt at a more complete list, the following elements are suggested as "desirable" methodology components, some of which are more "necessary" than others:

a) The System Development Cycle:

- Phases
- Tasks
- Milestones
- Guidelines
- Responsibilities

b) Project Acceptance/Approval Procedures

c) Milestone Documents

d) Milestone Document Approval Procedures

e) Cost/Benefit Analysis

f) Long-range Planning Interfaces

g) Project Management/Control Techniques

h) Estimating Techniques

i) Program Design Methodology

j) Data Base Design Methodology

k) Risk Analysis Procedure

l) Change Control Procedure

m) Hardware/Software Selection Guidelines

n) System Assurance Reviews

o) Post Implementation Reviews

These components of a development methodology are further described as follows:

a) The System Development Cycle or, as some may prefer, the Project Life Cycle, is central to a methodology.

For our purposes, such a "cycle" is meant to encompass all activities or occurrences from the inception of a development effort through implementation and operation of the completed system. It provides a familiar framework or structure within which application systems of different sizes and/or complexities can be developed by Project Leaders, Systems Analysts, and Programmers (or other appropriate titles for system developers). The System Development Cycle has its own components, as follows:

1. Phases are stages of development activity associated with or named for the type or level of activities being conducted at that time. These phases are not to be confused with sub-systems or projects which may be implemented in a "phased approach." Project life cycle phases have such names as Feasibility Study, General System Design, Programming, Implementation or other names which may be appropriate for the activities within the boundaries defined for the respective phases. You should avoid placing constraints on your project team by pre-naming

and restricting the number of phases as was done in the case study. You should remain flexible in this regard.

2. Tasks are the development type activities to which Toellner alluded. These tasks are separate and essential actions which should take place at a particular time or in a particular sequence in order to construct certain types of data processing systems. As Toellner points out-- "the life cycle controls the sequence"(4). Generally, tasks will be assigned to a particular phase within the System Development Cycle. In other cases involving different system applications, a particular task might not be performed in the same phase of the development cycle. In still other cases, a task might be started in one phase and completed in another.

David Katch, in Systems Development Methodologies - A Tutorial, (21), states that "The Standard Framework" should be sufficiently rigid so that the installation can benefit from a "standard and consistent" way of doing

all the development work, and to serve as a basis for effective project control. "But at the same time . . . one should have sufficient flexibility so that the methods can be molded to fit the given project." In other words, depending upon project size and/or complexity, certain tasks can be omitted while still operating within the "standard framework." Examples of development type activities or tasks (albeit, very general) which would be found in a methodology might include the following:

- Review the existing system,
- Define the problem,
- Estimate development manhours,
- Perform a cost/benefit analysis,
- Prepare user documentation,
- Etc.

3. Milestones, within the System Development Cycle, are status marking events or activities by which development progress can be measured. Such an event is usually the completion of a particularly "significant" development task. As P. W. Metzger states, in Managing A Programming Project, (3), each milestone should be based "on

something measurable, otherwise, you won't know when you get there." Their purpose, therefore, is to assist project leaders and managers in the performance of project control functions. Such a milestone activity can occur within a development cycle phase, or it can signify the completion of a phase itself.

4. Guidelines provide, as Toellner says, "the reasonable guidance to the staff on how to execute each of the tasks of the life cycle." Katch goes one step further and suggests that,

"To be effective such guidelines should not only guide . . . but they should also 'teach'. If the methods guidelines do not include adequate explanation of the underlying concepts and why certain design considerations are important, then the staff is not learning." (21)

This area provides the greater part of the methodology contents..

5. The responsibilities of project team members, users and data processing management should be defined for each task in which they participate, whatever the phase of development. As Katch

says, "Behavioral Scientists have always maintained that people respond more favorably if specific roles and responsibilities are clearly defined and each person knows what is required of him, when, and the standards against which his performance will be measured." (7)

b) Project Acceptance/Approval Procedures may be the next facet of your methodology which you will wish to address, particularly if there are many departments and/or users competing for data processing services. These procedures may establish the circumstances requiring approval for either initiating or continuing development work. Such circumstances, in turn, may relate to the size of the development effort, the life-cycle phase being completed, or significant changes in scope and/or effort required.

The procedures may also address the individuals, positions, or agencies responsible for authorizing the initiating or continuing of a development effort. They may also describe the preparation of documents, together with instructions for submission to and/or review by the authorizing agencies.

c) Milestone Documents, their format and requirements, should be described by your methodology. They are primarily forms, charts, worksheets, reports, and proposals which are prepared to document the results of milestone activities. They also serve as tangible evidence of what was performed as well as proof that the milestones had been reached and completed. The following are a few examples of "milestone documents":

- External Specifications,
- Logical data base design,
- Structure Chart,
- Acceptance Test Agreement,
- General System Design Proposal, etc.

The inclusion of this component is also recommended by Katch primarily for project control purposes:

" . . . one methodology consideration that is critical for achieving effective control is the employment of an end-product orientation for all tasks and phases. A tangible end-product is a clear-cut way to delineate something that is finished." (21)

Another similar argument is presented by E. B. Daly:

"Because the early phases of software development have no visible output except documentation, major milestones must be associated with a completed documentation package.

A defined milestone with no visible output is a useless milestone for management control." (12)

d) Milestone Document Approval Procedures are essential if your methodology is to achieve the "effective control" promised by adoption of an "end-product orientation." Appropriate approval agencies should be designated for the respective milestone documents required by the methodology. These agencies could be the system's future users, data processing technical or managerial personnel, or other members of the project team, such as the Project Leader or Project Manager.

Milestone Document Approval Procedures should describe how and by whom a document is to be evaluated, and what should be done with the document whatever the outcome of the review. A milestone document prepared for no other purpose than signifying completion of an activity or group of activities, must be reviewed and/or critiqued by an agency external to the Project Team if the preparation activity is to have any meaning or beneficial purpose.

e) Cost/Benefit Analysis consists of the identification of the following:

- current costs of the area of concern,

- the expected future costs of the area after implementation of the proposed system,
- sources of tangible benefits, and estimates of their amounts for both on-going (recurring) and one-time situations,
- intangible benefits,
- the actual costs to date and estimated future costs for development of the system, and
- a measure of the proposed system's return on investment (See Section 3.1.5).

Tangible benefits are those which provide cost savings or otherwise increase profits and include force reduction, productivity gains, reduced out-of-pocket costs, increased sales, etc. Quantifying the value of the information provided by the system may be an important sub-division of this component. Another sub-division may be the expression of these amounts as a range of returns, such as "lowest reasonable return," "best estimate," and "highest reasonable return" or some other form of sensitivity analysis so that Management can see the impact of changes in certain key benefit assumptions. Note that avoidance of greater operating costs is a valid economic return on investment concept and therefore, could be considered as a

tangible benefit. In this case, however, there must be some other compelling reason, which must be identified, for justifying the new system.

Intangible benefits are generally non-monetary benefits or those for which quantification is otherwise impossible. Any situation which tends to make a new system mandatory may qualify as an intangible benefit for the purposes of system justification. As such, intangible benefits may be more important than those classified as tangible.

Other aspects of the Cost/Benefit Analysis component may be its frequency of performance, its intensity or level of detail, and its presentation format, all of which may vary by life-cycle phase or project size.

f) Long-Range Planning Interfaces are of importance only if your data processing organization is required to engage in long-range planning. Interfaces can occur at the following points:

- Project Classification rules (e.g., Maintenance vs. small, medium, or New Development projects)
- Milestone Document Approvals
- Project Acceptance/Approval Procedures
- Project Accounting (Timekeeping)

For the purposes of these guidelines, long-range planning is a separate and distinct sub-set of a data processing organization's project management system. It is considered, therefore, to be external to the System/Project Development Methodology.

g) Project Management/Control Techniques can be spelled out by your methodology as a standard. For example, your methodology can specify or recommend using PERT or CPM as standard project planning tools or the use of Gantt or modified Gantt charts for reporting project status or progress. This may be desirable as a methodology component only if you find it necessary to use the methodology to legislate a standard. The establishment of standard milestones (tasks) and use of milestone documentation, as well as project monitoring (timekeeping), will also assist the project management function.

h) Estimating techniques may also be specified or referenced by your methodology if you believe you have found guidelines that can help your organization. In addition, your guidelines can specify when estimating is to take place and how estimates are to be used, for

example, as in developing PERT/CPM networks and in cost/benefit analyses.

i) A Program Design Methodology may be an important feature of your methodology particularly if a standard approach to design is advocated for the entire Systems and Programming Section or Department. Structured Design and Structured Programming are possible examples for this component.

j) A Data Base Design Methodology may be described in those phases to which the data base design task is assigned. This refers to both logical and physical design of the data base and is concerned with a more organized, structured approach to data base design than those traditionally practiced by system developers. An example would be Service Analysis, as described by L. T. Cohen in his book, Data Base Management Systems (22), and in the VAI Video Series of instructional material relative to data base design.

There is a danger in including this kind of formalized technique in your Methodology Manual. There is always the chance that a better technique will appear on the scene necessitating a substantial revision

to the manual. This has not yet occurred at the company in the case study, but the Service Analysis approach to Data Base Design is being restudied because of numerous complaints about the amount of time required to comply with its rules. This "potential pitfall" label also applies to other techniques such as Structured Design, Structural Programming, HIPO's, etc. Inclusion of this component in your guidelines would be desirable only if a standard approach to design is to be legislated for the entire Systems and Programming organization.

k) Incorporating a Risk Analysis procedure in the development methodology is a "gimmick" used by some organizations to force systems developers to consider at various stages of development the probability or likelihood that a proposed system will not meet its original objectives. Such objectives would be related to the system's functional specifications, benefits, costs, and development effort (manhours and calendar time). Risk Analysis procedures can be enhanced by including a questionnaire which supposedly will assist in measuring risk by assigning values to a variety of

responses. The questionnaire could address potential problem areas relative to a system's or project's size, structure, and technology requirements as defined by F. Warren McFarlan and D. Norton in their article on Project Management (23). Such a procedure and questionnaire were included in the case study's Methodology Manual in a special section for attachments.

4) Change Control Procedures are those which intend to regulate, if not reduce, the number of changes to the system specifications requested over the remainder of the development cycle. Such regulation may be necessary:

- to guarantee that all changes are addressed formally,
- to allow system developers to determine the impact of changes on manhour and duration estimates for system development,
- to guarantee that each request for change represents the consensus of the user department, and
- to assure that the user department recognizes and accepts the responsibility for additional charges and/or delays caused by release of a request for change.

This component may have documentation requirements in addition to procedures, all designed to determine the

relative value of the change in terms of the impact on the development effort, the impact on the total shop's development plan, and the impact on the achievement of the system's objectives.

m) If Hardware/Software Selection Guidelines are included as a component, their intent would be to improve the evaluation of alternative solutions so that subsequent recommendations would be sound. Such guidelines would serve as a methodology within a methodology by establishing tasks, task performance sequence, and responsibilities for task performance. In addition to a liberal sprinkling of technical review sessions, these tasks might include the following:

- Develop Design Goals
- Review for Feasibility
- Develop Internal Specifications
- Develop Evaluation Questionnaire
- Develop Request for Proposal
- Evaluate Responses
- Select a Vendor and Model

This component was also included in the attachment section of the case's Methodology Manual.

n) System Assurance Reviews are basically of two types, technical and managerial. E. B. Daly, in Management of Software Development, states that "managerial reviews

are held at major development milestones" for the purposes of reviewing milestone documents, evaluating project costs and schedules, and approving or disapproving continued development. He also indicates that "technical reviews are much more detailed than management reviews and do not consider schedules and budgets." (12) Daly also claims technical reviews are "to analyze the same commercial documentation presented to management, but in a very detailed manner" and that technical reviews should be completed before such documentation is submitted to management.

The author agrees that milestone documents (of whatever sort) should be reviewed by qualified technical personnel prior to their release to management, particularly user management. On the other hand, the technical review is performed to insure that the document's quality meets the expected departmental standards and that all required aspects of the development methodology have been fulfilled. Therefore, such technical reviews should also apply to "schedules and budgets." As it involves the control of the quality of the entire system development effort and its product,

the system itself, this component is one of the more important "needed elements" of your methodology. It may serve "to give the needed authorities and to ensure that the process is used." (4)

o) Post Implementation Reviews are evaluations of installed systems after their users and operators have had some experience with them. Such reviews might also evaluate the development period of those systems in order to help improve development practices on existing and future applications through effective updates of your methodology. The Post Implementation Review (PIR) should help to establish the credibility of the user departments with respect to benefits claimed and should also affirm the credibility of the data processing organization with respect to manhours and duration estimates. Nevertheless, the primary objectives of the review are the following:

- Determination of the degree of successful achievement of the project's objectives (measurement of a system's value),
- Determination as to whether or not justification was and continues to be supportive of the project's expenses,

- Evaluation of the project's planning in order to make recommendations concerning revisions to future planning activities,
- Identification of problems and resolution of any remaining difficulties regarding the application of previously contracted services,
- Evaluation of the project's documentation as to completeness and compliance with standards, guidelines, and/or policies.

According to a recent, limited study by the Diebold Group, New York, "Most organizations do not understand how to measure the MIS activity, (nor do they understand) how it performs effectively." According to their study, only 8 of 33 respondents planned to use Post Implementation Review as "the last step in the development cycle of a DP project to ensure the project makes a significant contribution to the company's ability to attain its goals." (24) The mail survey conducted for this thesis revealed that only 15 of 40 respondents claimed to utilize the PIR as a methodology component. It was also interesting to note that, of the five organizations who felt that their methodologies were not successful, four did not employ the Post Implementation Review.

Of the fifteen possible methodology components mentioned above, most were utilized by a majority of the respondents to the thesis questionnaire who claimed adherence to a development methodology. The major exception was Risk Analysis for which only four of twenty-five claimed inclusion of such a procedure. Long-range Planning Interfaces, Hardware/Software Selection Guidelines, and Post Implementation Reviews were claimed by only 60% of the "methodists."

3.2.10 Write a Specifications Document

After considering the components comprising your development methodology, the project team will have reached its first major milestone. The first phase of the project will have been almost completed. The team should now prepare a "milestone document," i.e., specifications for the development methodology manual. These specifications could also be called user requirements and would not just list, but would describe the contents and thrust of the manual much like the "white paper" of the case study.

Completion of the specifications document would place the project team in a better position to proceed

to the next step toward implementing a System/Project Development Methodology. Whether it is the review of vendor/consultant packages or the writing of the manual in-house, your organization would now know what it is they want in their development system.

Chapter Four

MAKE VERSUS BUY CONSIDERATIONS

4.1 Introduction

An approved specifications document will define your expectations for a System/Project Development Methodology. At the same time, it will delineate the scope of your effort. With your requirements firmly established you will be in a much less vulnerable position when advancing to the next problem level or phase of development. Having already established that you intend to implement a methodology, the next major hurdle is the decision as to whether it will be designed and written in-house or purchased.

If your management has already opted to purchase a particular commercial version, then only those guidelines relative to implementation (see Implementing Your Methodology, Chapter 6.0) may possibly apply. This limitation derives from the fact that these guidelines are oriented toward in-house development. If Management has preordained that the methodology should be developed in-house, then it might be expedient only to estimate

manpower and duration requirements as well as costs before continuing with the production aspects of the Manual. If your management desires a more objective analysis before making their choice, then one should continue by pursuing the following tasks:

- Identify existing components.
- Review commercial packages.
- Establish interface requirements.
- Estimate manpower and duration requirements.
- Perform a cost/benefit analysis.
- Select an alternative.
- Write and submit a proposal.

4.2 Identify Existing Components

Begin your analysis by reviewing your organization's existing development procedures, policies, forms, and milestone documents in light of the selection criteria in your specifications document. Make certain that you know, before proceeding any further, your inventory of methodology components. Then note those items, perhaps forms or guidelines, which can be adapted to your use from other, non-proprietary sources. The remaining

requirements must be fulfilled by either in-house production or obtained by the acquisition of a commercial methodology. With this exercise, you will have made it easier to identify those features offered by another methodology which are either already available, still required or otherwise unsolicited.

4.3 Review Commercial Packages

There are over 50 commercial Project Management System products available. However, they are concerned with different aspects of project management, for example:

- Manual-structured systems (Methodologies)
- Project tracking systems
- Project networking systems
- Full PMS or a combination of the previous three (14).

As these guidelines are concerned solely with System/Project Development Methodologies, one must be careful to isolate those which treat the subject with which we are concerned.

There are several sources from which you can obtain information about various commercial methodologies

currently available. In addition to advertisements in periodicals devoted to the "automatic information handling" industry, product descriptions or references to commercial packages can be obtained from at least the following two sources: EDP Analyzer, a Canning Publication, 925 Anza Ave., Vista, California 92083 (Request Volume 14, No. 9, and Volume 17, Nos. 1, 2, and 3); and ICP Interface, a Software and Data Services periodical published quarterly by International Computer Programs, Inc., 1119 Keystone Way, Cornel, Indiana 46032. From the summary data displayed in these sources you can select those packages which appear to offer what is desired. Usually, information is provided for obtaining additional promotional material.

You will find that the assertions made for these commercial products in the promotional literature are complex and possibly bewildering. As discovered in the case study and as found in the literature review (see Chapter 2, Section 2.2), the promotional brochures provide "some insights into the components of (the) respective methodologies," but it is the author's opinion that they do not tell enough about the package to grasp

a thorough understanding of its contents. Based upon the experiences of the project team of the case study, it would appear that vendors marketing these products tend to guard their secrets jealously. Apparently, they fear their offering will lose some of its marketability if its contents are viewed by the potential customer before the sale is consummated. Nevertheless, in addition to a personal contact with the vendor, you must insist on viewing the contents of the commercial methodology without any obligation to buy. Remember, you are not yet obligated to purchase any package. Just as it happened in the case study, you, too, may be required to sign some sort of non-disclosure agreement. If this occurs, your Legal Department and Purchasing Agent should review such agreements before they are signed in order to protect your employees' and your company's interests. Members of the project team in the case study were able to conduct their review of the proprietary methodology without fear, knowing that the agreement they signed protected them as well as the vendor.

Given the opportunity to study the contents of a methodology, the reviewer should compare the components and features of the offering against the specifications which had been prepared previously. Keeping score can serve to measure the suitability of the package for your purposes. It can also assist in ranking the various products under consideration. It is also recommended that any "nice-to-have" features of each prospect which are not listed in your specifications be noted as well. This type of information can serve as a tie breaker in case two or more products are of equal rank. Three other suggestions: First, take at least one entire day and no more than three days to review the vendor's material. Second, refrain from making a commitment to any vendor until your investigation has been completed. Third, as stated in Chapter 3.0, Section 3.2.7, do not visit a company that purchased a proprietary package in order to evaluate its system. However, if a decision is ultimately made to purchase a particular package on its merits, contact should be made in order to establish references for the vendor as to their reliability and performance. At the same time, data relative

to the user's customizing experience (manhours, duration, etc.) should be obtained for evaluation purposes.

It is not the author's intent to suggest any additional evaluation criteria to include in your specifications other than the concepts, environmental factors, and components mentioned previously. A more complete list would define a methodology that would be satisfactory to the author, but it would not necessarily be suitable for the reader's organization.

4.4 Establish Interface Requirements

Every commercial methodology will be different and each would probably require modification to adapt it to your particular organization. Modification would more than likely occur at points where a new methodology would interface with existing practices which you may wish to retain. Forms will also have to be revised to suit local terminology and data requirements. Attempt to develop a list of tasks relative to each different modification activity which may be required. Such a task list will be used to develop cost and duration estimates for the various alternatives. A word

of caution: Do not consider implementation costs at this point in time. Implementation and training are activities which must be performed whether the methodology is purchased or developed in-house.

4.5 Estimate Manpower and Duration Requirements

Estimates of effort and duration must be prepared for each alternative including that of in-house development. Estimates will vary in proportion to the amount of new writing or modification which may occur. Based on the experience of the company in the case study, the following might be considered a heuristic for estimating effort and duration for writing a Methodology Manual with several life cycle phases and in considerable detail:

• Planning	2 weeks
• Research	1 month
• Preliminary Draft of Specifications	1 month
• Debate, Review, and Write Specifications	1 month
• Make or Buy Study	1 month
• Proposal Approval	1 month
• Planning	2 weeks
• Production and Implementation	1 month per life cycle Phase or separate guideline
• Education & Training	1 month

of caution: Do not consider implementation costs at this point in time. Implementation and training are activities which must be performed whether the methodology is purchased or developed in-house.

4.5 Estimate Manpower and Duration Requirements

Estimates of effort and duration must be prepared for each alternative including that of in-house development. Estimates will vary in proportion to the amount of new writing or modification which may occur. Based on the experience of the company in the case study, the following might be considered a heuristic for estimating effort and duration for writing a Methodology Manual with several life cycle phases and in considerable detail:

• Planning	2 weeks
• Research	1 month
• Preliminary Draft of Specifications	1 month
• Debate, Review, and Write Specifications	1 month
• Make or Buy Study	1 month
• Proposal Approval	1 month
• Planning	2 weeks
• Production and Implementation	1 month per life cycle Phase or separate guideline
• Education & Training	1 month

As reported earlier, the methodology writer in the case study contributed the equivalent of one man-year of effort. This contrasts with the times reported by the mail survey respondents who had installed commercial methodologies. Of those five, the least amount of time and effort reported for modification and implementation was an elapsed time of one year involving two man-years of effort.

4.6 Perform a Cost/Benefit Analysis

The Cost/Benefit Analysis suggested here may not be the usual variety recommended for the justification of an information system. It may take that form if the completion of the project, i.e., development and implementation of a System/Project Development Methodology is at stake. If the methodology must be justified, the 'benefits' to be identified may be either tangible or intangible and should result from the data processing organization's adherence to a formalized, structured approach to system development. Costs of the methodology, in this case, represent those of the best alternative solution, either in-house development or

purchase of a commercial product.

In situations where the installation of a methodology has already been justified, for example, as occurred in the case study, it is only necessary to present sufficient information to enable management to select the best alternative solution. It is solely the 'make or buy' decision which we are concerned with here.

When estimating the costs of in-house development, consider the following activities encountered in the case study subsequent to the 'make or buy' study:

In-House Development

Case Study Experience

Draft writing & form design	*
Typing of Drafts (double spaced)	400 pages
Proofreading	400 pages
Editing	*
Retyping draft corrections	20 pages
Duplication and Stapling	400 pages x 25 copies
Review responses and make changes	*
Write summaries of action taken	*
Retype Final Copy (Proofreading insignificant)	400 pages
Duplicate, Punch and Staple	400 pages x 150 copies
Mailing expense	(company mail)

* 830 (of total of 1900) man hours @ \$22/hr. or \$18,260.

When developing costs for purchasing and installing a commercial package, the case study's project team

estimated that it would take the same amount of expense for forms printing, management reviews, implementation (including education and training), and general follow-up time for both alternatives. For that reason, these activities were excluded from the previous list. However, there will be additional costs for new writing, revisions to the package, and redesigning of forms. There may or may not be any duplication costs depending on whether sufficient additional copies of the methodology are supplied by the vendor at no charge. Although the above costs will vary with the amount of revision necessary, vendors of commercial methodologies concede that the list price of the package is only a part of the total costs which will be incurred (25).

A sampling of some of the products available and their prices are shown below. (It is not known if the listed prices permit distribution to multiple sites):

<u>Product Name</u>	<u>Vendor</u>	<u>1978 Prices</u>
SDM/70	Atlantic Software	\$17,500
PROVAC	Software Brokers, Ltd.	\$17,500
SPECTRUM-1	J. Toellner and Assoc.	\$25/30,000
SPECTRUM-2	J. Toellner and Assoc.	\$10/18,000
SYSTEMACS	Management and Computer Services, Inc.	\$30,000
PRIDE	M. Bryce & Assoc.	\$18,000
CSL	CAPSOGETTI's LOGICIEL	\$100,000

4.7 Select An Alternative

Determining the best alternative involves the exercise of judgement to a large degree, particularly when considering advantages and disadvantages of an intangible nature. In other words, the best alternative may not be the least-cost alternative. When evaluating the pros and cons of the two choices, one should estimate their relative impacts on the following:

- Speed of implementation (project duration)
- Amount of disruption in system development, operations, employee morale, etc.
- Perceived quality of the guidelines
- Completeness of the methodology with respect to the original specifications
- Amount of documentation proposed (insufficient vs. excessive)
- Cost

- Use of cash as opposed to working capital
- Ability of the methodology to manage system development
- Ability of the methodology to generate good systems.

The company of the case study obviously selected the in-house alternative. Management was largely influenced by the high licensing fees charged by vendors, which at the time of the study were significantly higher than the prices listed in the previous section. The company was not in the best cash position at the time, and considering the fixed cost nature of the project team, elected not to spend the cash. The management was also influence by the following:

- Confidence in the writer based on the quality of the specifications document.
- The ability to implement the methodology in steps as each part of the manual was approved.

Guidelines for the early phases of development were the most critical. With phased implementation, the company could begin to reap the benefits of these particular sections early in the cycle, in spite of the protracted schedule for in-house development. Based on the above criteria, you should be able to select an alternative

which would be the 'best' for your particular situation.

4.8 Submit a Proposal

After the project team has completed its analysis, a proposal (or a report with recommendations) should be prepared. Such a report will serve as a phase-end document, and as such, it constitutes a significant project milestone. The document will be submitted to data processing management for review and approval. Subsequently, it can be used by the project team as an aid to understanding all aspects of the project known to date. Therefore, use care in the development and presentation of the material in the report.

The report need not be wordy or voluminous. However, it should have information organized in a good presentation format. The following table of contents is recommended:

- a) Title page or Cover letter
- b) Table of Contents
- c) Management Summary
- d) Statement of the Current Situation or Problem
- e) A Statement of the Objectives of the Analysis
- f) A Statement of the Scope of the Analysis
- g) Alternative Solutions

- h) The Proposed Solution (including a discussion of its requirements, advantages, disadvantages, and reasons for its selection)
- i) Cost/Benefit Analysis
- j) Preliminary Implementation Schedule
- k) Project Manhour and Duration Estimates
- l) Comments and Recommendations

The report can be used to bring any significant development problems to management's attention, such as the impact of the system on individuals and the organization. If the project team believes suspension or cancellation of all or part of the remaining development effort is in order, it is their responsibility to make that recommendation and emphasize any of the proposal's shortcomings in the management summary. If the team recommends continuing the effort via in-house development, the remaining chapters of these guidelines would be helpful in bringing the project to a successful conclusion.

Chapter Five

GUIDELINES FOR DEVELOPING YOUR MANUAL

5.1 Introduction

Having made the choice to develop your Methodology Manual in-house as a consequence of the 'Make or Buy' study, future success may depend upon continued adherence to the principles which were proposed earlier in Chapter 3, Sections 3.2.1 through 3.2.5. The development effort was to have been considered as a project. The problems, objectives and scope of the effort were to have been defined. A development team was to have been assigned. A methodology was to have been applied, and a project plan was to have been formulated, at least for activities preceding the production phase of this particular project's life cycle. If these steps were not followed, it is imperative that they be attended to now before proceeding.

5.2 Provide for Project Control

Project control usually involves the development of a plan of action followed by monitoring performance against the plan. It is suggested for the in-house

development project, that an individual, preferably a departmental manager of higher rank than the methodology writer, be assigned as Project Manager. Although he will be charged with project control responsibilities as opposed to quality and design responsibilities, his primary duty would be that of enabler. In this capacity he will shield the methodology writer from distracting influences, notably assignments from management or activities that could more easily be performed by someone else. In the case study the methodology project experienced considerable delay because of demands on the writer's time by management. At the same time, assignment of the more mundane tasks, including those related to project control, to personnel other than the writer helped speed the overall project. An objective manager, someone with clout, could provide both of these advantages for your project.

5.3 Plan For In-House Development

Another duty of the Project Manager will be to participate in the project's planning. First-hand knowledge of the tasks to be performed, by whom they

will be performed, in what sequence, and in how much time, will assist the Project Manager in the monitoring of the project as soon as production begins. As suggested earlier, an activity arrow diagram such as a PERT chart should be used to establish the project's critical path. Identification of and removal of as many non-critical tasks as possible from the project's critical path is part of the necessary planning. It could even shorten the project's duration as occurred in the case study.

At this stage, the manual builders have in their possession an additional tool which had not been available earlier, an approved specification document. The system's "information requirements" have been defined and reside in that document. It is the project team's responsibility to design and produce the system (the Methodology Manual) based upon the specifications, unless they had been revised as a consequence of the 'Make or Buy' study. However, before beginning to write, the specification can be used as a basis for performing the necessary planning. The following steps are a suggested approach:

- Establish an implementation strategy (see Chapter Six)
- Construct a task list compatible with the specifications and the implementation strategy (Also see Section 5.6)
- Establish a review and approval procedure based upon the above and insert it in the task list (See Section 5.4)
- Insert a task for establishing documentation standards (See Section 5.5)
- Assign responsibilities for the various tasks to team members
- Estimate manhours and elapsed time for each task
- Determine predecessor-successor relationships and assign task sequences
- Establish the project's critical path by means of the activity arrow diagram or PERT chart mentioned above
- Attempt to optimize the critical path by utilizing the project team's resources as efficiently as possible

- Identify the project's various milestones, preferably those having a product associated with it, for project control purposes.

If the Project Manager and the team are satisfied that they have established a reasonable plan and schedule, they should proceed to execute the plan.

5.4 Establish a Review and Approval Procedure

The adoption of a review and approval policy with an appropriate procedure is a necessity for ensuring the success of the in-house venture. A critique of the Methodology serves several purposes, just as a system assurance type review does (see Section 3.2.9, component 4). It is performed to ensure that the document's quality meets the expected departmental standards and that all required aspects of the specifications have been fulfilled. By performing a system assurance review you have recognized the "tendency to err, (a) viewpoint (that) recognizes that mistakes of both omission and commission are sure to occur at all stages of system building." (26) Not all criticism will be either valid or good, but the methodology will

rarely be hurt by a "second opinion." If the material submitted for review represents a complete unit of work such as a Section of the Manual, it meets the definition of a "milestone document" (see Section 3.2.9, Component c). As such, it provides "tangible evidence" that a milestone has been achieved and permits a measurement of progress on the project.

The review process also serves to involve more members of the staff in the methodology development effort and should serve to alleviate resentment on the part of the System development staff toward the new methodology. Examples of this resentment, sometimes called the NMH (not-made-here) syndrome, were encountered in the case study. The corporate group responsible for development of the Methodology Manual recognized this possibility and attempted to neutralize it by requiring participation of the divisional staff by means of reviews and constructive criticism. If this approach does not work, a combination of salesmanship, cooperative effort, strong central control and enough confidence to persevere in the face of opposition may be required to muscle the project through to completion.

The experiences of the case study also illustrate how not to conduct a review of the methodology. Considerable energy, management time, and project team time can be wasted, if reviews are conducted with data processing and other management in a meeting-type format. The reviewers should take reasonable time to study the material in private prior to asking questions or commenting. Furthermore, all comments and suggested changes should be submitted in writing and signed by the reviewer.

Who should serve as a reviewer? Who should pass judgement on the concepts and policies expressed by the methodology? Who will determine that those concepts and policies represent the thinking of the company's management? Such judgements can only be leveled by qualified technical and managerial personnel who can recognize whether the guidelines are reasonable or not and can then agree to abide by the manual's instructions and direction. The reviewers are in reality also the approvers of the methodology whether they perform that function directly or indirectly. It is recommended, therefore, that the responsibility for

review and approval reside with the Manager of Data Processing, and the heads of the various departments within the Data Processing organization. This approach worked well for the company in the case study. In that example, there were instances when a manager delegated the review to Supervisors of Systems and Programming and/or respected senior development personnel. There were no significant differences in the quantity or quality of responses from managers who delegated reviews as opposed to those who did not. An organizational consensus was achieved in either case.

After designating the appropriate review and approval agencies, you should establish rules as to how each part of the manual should be evaluated, how much time the agencies have to do it, how responses are to be processed, and what should be done with the methodology material following the review. In the case study, reviews were conducted on each separate section of the manual after the typing of its first draft. Reviewers were required to respond with their comments by a fixed date (usually 2 weeks from distribution of the draft). Comments and suggested changes were then

reviewed by the project team, and those accepted were incorporated in the final draft of the manual. Having decided what your review and approval procedures are going to be, disaggregate the steps, and insert them into the project plan in the appropriate location.

5.5 Establish Documentation Standards for Your Manual

One of the first operations that you should perform under the new plan is the adoption of documentation standards. This is not absolutely necessary, but you will hate yourself later if you do not help yourself now by addressing the following:

- Organization, Format, and Content
- Pagination
- Glossary
- Standard Documents

The writing task for the Methodology Manual in the case study proceeded much more smoothly after decisions were made relative to documentation standards. It is interesting to note that, although the outline for the Methodology Manual contents occasionally underwent change beyond the current point of writing, no organizational changes were made once a section had been written and published.

5.5.1 Organization, Format, and Content

In the case study, it was the data processing organization's practice to assign a single alphabetic character to each procedural manual such as the Methodology Manual. This alphabetic character would precede all other characters identifying various chapters and sections. Any reference, therefore, to a particular chapter or section would also identify the manual it came from. Organizationally, each major division was referred to as a Chapter. Each Chapter would be assigned at least a 2-digit number in multiples of 10. That way additional chapters could be inserted after several had been completed. Assume the Manual was assigned a letter 'M'. The first chapter, therefore, would be designated as "M 10" and followed by its title. The next major sub-division of a chapter was referred to as a Section and was also identified by a two-digit number for the same reason. Subsections were identified by Roman numerals. The Manual, accordingly, was organized as follows:

M 10 (Chapter 1 TITLE - all capital letters)

.10 (Section 1 Title)

I (Sub-section I Title)

A. Subsequent divisions

B. etc.

II (Sub-section II Title)

.20 (Section 2 Title)

References to Chapter 1, Section 1, Sub-section I would then appear as Sub-section M 10.10.I" without having to spell out the complete description.

Some of the titles assigned to various chapters and sections in the "System/Project Development Methodology" published by the company in the case study (26) read as follows:

M 10 INTRODUCTION

.10 Purpose and Use of the Manual

.20 Maintenance of the Manual

M 20 GENERAL OVERVIEW

.10 Project Life Cycle

.20 Relationship of Sub-systems, Project and System

M 30 (FIRST LIFE CYCLE PHASE), etc.

.

.

.

M 90 (LAST LIFE CYCLE PHASE)

M100 ATTACHMENTS

.10 Hardware/Software Selection Procedures

.20 Risk Analysis Questionnaire

M110 DOCUMENTS USED AND PRODUCED

The format and content of each chapter dealing with a life cycle phase was then regimented as in the example

below:

Mn 0	Phase Title
.00	Table of Contents
.10	Introduction
.20	Phase Objectives
.30	Phase Responsibilities (Individual functions - by Department)
.40	Documents Used (Within the Phase; a summary)
.50	Documents Produced (Within the Phase; a summary)
.60	Phase Activity Network
.70	Phase Activities and Guidelines
	I. Develop a Phase Plan
	A. Definition (Guidelines)
	B. Documents Used and Produced (within an activity)
	C. Organizational Interfaces
	II. etc.

The above format is an adaptation of one employed by Monsanto Company for their Methodology Manual (27).

This particular format was selected by the case study development team and was followed faithfully through to completion of the manual.

5.5.2 Pagination

Defining the pagination rules was relatively easy once the Manual's organization scheme was resolved. Pages were to be numbered in much the same way as the manual was organized. Each page number began with the Manual letter followed by the Chapter number, followed by a period, followed by the Section

number, followed by a period, followed by a two-digit sequential page number which began at .01 for each new section. Therefore, the first page of the Phase Activities Section of the First Life Cycle Phase was numbered, e.g., M30.70.01. Revisions to the Manual requiring inserts were suffixed with lower case letters, e.g., M30.70.01a, etc. All page numbers were positioned along the bottom margin in the right-hand corner.

Other rules and detail related to pagination employed in the case study's manual included the following:

- The "release date" was typed in the lower left-hand corner of each page
- The Manual title, "SYSTEM/PROJECT DEVELOPMENT METHODOLOGY" was printed together with the company's logo in the top center of each page.
- Every new sub-section began with a new page.
This facilitated revision of the Manual without endless retyping.
- A heading was typed on each page denoting the appropriate Chapter and Section identification numbers and titles.

5.5.3 Glossary

You may wish to set aside a Chapter in the Manual for a glossary. This may be desired by the user community as protection against the excessive use of computerese of which the data processing business has historically been accused.

In the case study, production of the Glossary held the lowest priority, but, as in documenting a computer system after the system had been installed, it was never completed. Unlike System Documentation, however, it was subsequently felt that it was not worth the effort, partly because the Methodology writer practiced what he preached--avoidance of excessive usage of data processing jargon.

5.5.4 Standard Documents

You may have noticed in the examples of organization and format discussed in Section 5.5.1, a manual chapter entitled "M110 DOCUMENTS USED AND PRODUCED." The Methodology writer in the case study used this device to introduce standard forms and standard report formats which were referenced within the Guidelines. This applied to both optional and required

usage. It also succeeded in achieving a uniformity of forms use between and within the various data processing installations. Each new form or report was identified as if it were a "Section" within the M110 chapter, but with sequential numbers. Appropriate sub-sections were inserted as they applied to the particular document, for example,

- .01 Document Name
 - I Purpose
 - II Significance
 - III Instructions for Preparation
 - IV Sample Form or Report Format
 - V Instructions for Use
 - VI Example of Completed Form
 - VII Distribution
 - VIII Approvals

Establishing documentation standards such as these early in the project will speed the writing process because of their built-in outlining capabilities and project control milestones.

5.6 Logistics

There are several operating details which should not be overlooked. It would help to consider such mundane tasks as writing, typing, proofreading, correction, reproduction and preparation, distribution and mailing, editing and updating. The following sections describe

how these operations were handled in the case study.

5.6.1 Writing

The chief writer of the Methodology fortunately had duties other than the composing of sentences. Research of a large variety of subjects was required, and the revision of old forms and design of new ones helped to relieve the tedium of composition. Nevertheless, in order to use his time most productively, tasks more easily conducted by someone other than the writer were made the responsibility of an individual at a "control desk." This person's responsibilities included the expediting and control of the flow of the writer's output through the "mundane tasks" listed above. In the case study, the Project Manager arranged for these duties to be performed by the supervisor of the Standards function.

As far as the actual writing is concerned, eliminate as many technical terms (jargon) as possible. Use plain, but succinct language. Good grammar is required. Wordiness is to be avoided. System developers must be able to find guidance in the Methodology, not endless drivel.

5.6.2 Typing

The case project was fortunate to have access to word processing equipment. This facilitated the typing of revisions as well as the approved final version. The Manual's pages were recorded on magnetic tape cassettes and filed for future changes.

Other suggestions relative to typing or preparation of original copies include:

- Maintain consistent use of the same type font.
- Maintain consistent margins.
- Mark revisions. Draw a light vertical line along the left margin as far as the changed material extends.

5.6.3 Proofreading

Have the individual at the control desk perform or arrange for proofreading and correction. The methodology writer should be interrupted only for questions relative to interpretation or context.

5.6.4 Reproduction and Preparation

The same control person should arrange for duplicating the appropriate number of copies of the draft and the final version as chapters are released. Material should be collated and stapled for distribution.

In the case study, copies of both the draft and final version had drilled holes in the paper suitable for use in 3-ring binders.

5.6.5 Distribution and Mailing

Because the manual was produced and released chapter by chapter, draft copies were distributed and reviewed at the same time as final versions for other chapters were being mailed and implemented. In addition, the distribution was not the same. Therefore, separate approved distribution lists were prepared and maintained by the control person. Standard cover letters were prepared advising those receiving copies as to the packet contents and appropriate instructions for responding or implementing.

5.6.6 Editing

In the case study, the chief writer was responsible for writing all changes and for editing the work of other members of the project team. Because of time restraints, however, only flagrant grammatical errors or overly wordy statements were changed. The chief writer should not be trapped into rewriting this material over a difference in writing style.

5.6.7 Updating

The highest priority activity of the methodology writer was the resolution of responses to the various chapter drafts, because as soon as all changes to a chapter draft were agreed upon, revisions could be typed and the chapter released for implementation. Explanations of reasons for rejection of every requested change were submitted to the control desk for compiling a summary of draft responses with their respective disposition. Summaries were then mailed not later than distribution of final copies.

The above examples of how the case study's project team handled some of the operational details prior to implementation should give you insights into the problems you will face as well as some possible solutions.

Chapter Six

IMPLEMENTING YOUR METHODOLOGY

There are obviously many ways one can go about implementing a development methodology. Much depends upon the number of "components" which were selected for inclusion in the manual or upon one's interpretation of what a development methodology is. EDP Analyzer, a data processing publication, is of the opinion that methodologies are meant strictly to govern the "design" process. They recommend installing the methodology in the same sequence as various techniques are learned. Since a company's learning sequence would be, "first, structured coding, then structured design, then structured analysis, and finally successive decomposition," . . . one should "start with the coding aspect and then work up toward successive decomposition." (28) However, a methodology with a broader view of systems development may require a different view of implementation.

The guidelines which we are recommending are those which provide a framework for the development of sys-

tems. Design of the system and coding are only activities, although important ones, within such a framework (see comments in Section 3.2.9, Consider The Components of Your Methodology Manual, relative to design techniques).

6.1 Phased Implementation

In the case study, such techniques as structured design were already being used but not universally within the company. One of the purposes of the development methodology, therefore, was to use the manual as a platform to promote (legislate) the adoption of preferred techniques throughout the company. Another was to establish where and when these techniques were to be used within the development cycle. The case company, however, had higher priorities which were mentioned earlier. One of their reasons for selecting in-house development was the speed with which the guidelines for the "early" phases of the development could be implemented, those being the most critical. So they chose to implement their manual on a chapter by chapter basis, each development cycle phase having its

own chapter. The chapters would then be released and implemented as a system might progress through the development cycle. First, however, an Introduction was installed followed by a General Overview chapter. These two chapters set the stage for what was to follow.

Implementing your manual in this fashion has some advantages. As mentioned previously, the methodology development would then have some built-in, bona fide milestones for project control. Another advantage, at least to a conservative organization, would be that the change in the way one conducts one's business would be a gradual one. Implementation, in the case study, was simple. It involved only the release and distribution of an approved final version of a particular chapter and any new documentation requirements generated by it. Instructions governing its use and authenticity came from the head of the organization and were issued well before the first chapter was released. Every Manager of Systems Development, Project Leader, Systems Analyst, and Programmer was required to read it and to follow its advice and direction at first

opportunity.

6.2 Provide for Education of All Staff Members

The case company's Managers of System Development were responsible for educating their respective staffs in the purpose and use of the Manual. Having participated in the development of the Manual through reviews and constructive criticism, they should have had a better idea than most staff members about what the purpose of each chapter was. Such an approach may be a good one; it may have worked for that company. It does appear to the author, however, that this particular situation is like the proverbial saw about leading a horse to water. In this case, however, with remote locations involved, the home office did not even know if the horse was led to the water, let alone whether he drank it or not.

The company had intended that the Methodology writer should visit all locations after the manual was completed for the purpose of assisting those who had questions on its application. Unfortunately, he was transferred to other duties before that plan could be

implemented. Nevertheless, this type of follow-up appears to be a rational approach. A personal appearance by the writer or by any management person who actively supports both the purpose and contents of the methodology should help improve its credibility. Two-way communications should be encouraged. Besides, any methodology can be improved and should be altered as weak spots are exposed. A classroom approach whereby many personnel can be reached in a short period of time should be considered. Those who must work every day within the rules and guidelines of the methodology should have the same opportunity for instruction and explanation as management and the user community. The success of the methodology is dependent upon the attitude of the development staff. If there is general cooperation, the personnel, department, and their systems will thrive. If the methodology is force-fed by an invisible home office staffer, it is the author's opinion that the potential for failure of the methodology and employee dissatisfaction will grow.

6.3 Provide for Follow-up

Providing for follow-up could be considered the same as providing for education. They both require manpower, that is, time and effort. What is meant here is more in the order of Systems Assurance Reviews. Someone should review documentation, reports, proposals and other phase-end documents in order to establish that procedures are being followed, in spirit as well as in practice. Reviews, with their opportunity for constructive criticism can provide an educational service as well.

The project team and other staff members performed this service for a while. There were some encouraging signs, and there were some distressing ones as well. There was the case of the Project Leader who wrote a Phase-end Report because he was told that he had to at the end of a particular development phase. Upon review of the report, it was apparent that 1) he had not read any part of the manual, and 2) his project was nowhere near completion for that phase. In another example, a clever Systems Analyst who was a free spirit, and disdainful of what, to him, was unnecessary paperwork,

overdesigned a system. His aim had been to discredit the methodology by proving that the documentation requirements took too much time and could not prevent bad systems from being proposed. One year later, the system was successfully installed, and the analyst was promoted to Senior Systems Analyst. The moral of the first story is, "Make certain everyone has access to a copy of the Manual." The moral of the second could be that a Systems/Project Development Methodology which leads to the successful implementation of systems is really beneficial. In addition, the latter also means that assigning qualified personnel with responsibility for system assurance reviews (see Component n in Section 3.2.9) and commensurate authority to reject material of poor quality is a more effective way of ensuring implementation of well-designed systems. Management must show signs that they are interested in the success of the Methodology by enforcing its directives and dealing constructively with transgressors.

6.4 Provide for Maintenance of the Methodology Manual

In the case study, someone was assigned the on-going responsibility for making changes to the Manual as the need arose. Occasions for revision have since been rare, and a full-time position has not been required. Nevertheless, the assignment was and should be made. In addition, the manual itself should direct those interested in revising the manual how to request such a change. Reserve a section of the first chapter in your manual to provide that direction.

6.5 Establish a Post Implementation Review

Just as you might wish to perform a Post Implementation Review (PIR) on an installed data processing system, so should your Methodology project be reviewed some time after it has been completely implemented. After the last chapter has been distributed, schedule a PIR at some future time in order to commit available resources to the review as well as to provide sufficient time to gain some experience with the Methodology. If your Manual has provided guidelines for this component, attempt to follow its direction as far as it is applicable. Consider these ideas for conducting,

the PIR:

- Someone other than the Methodology writer should coordinate and perform the review in order to provide a more objective analysis.
- Prepare questionnaires which address achievement of project objectives and problems related to the manual's performance. Submit them to interested personnel in order to save time and to be certain that specific points will be addressed by the appropriate parties before being called together for a review meeting.

This particular Post Implementation Review will have two major objectives, namely:

- To determine the degree of successful achievement of the project's objectives as previously stated in the "Make or Buy" study or other proposal. This includes an analysis of the project's cost.
- To identify in a formal way any problems relative to the Methodology.

At the conclusion of the Post Implementation Review, one can better gauge how well the Methodology is performing its function, and take appropriate steps to resolve any problems.

Chapter Seven

CONCLUSIONS AND AREAS FOR FURTHER STUDY

7.1 Conclusions

There are several points relative to system/project development methodologies in general and in-house guidelines in particular which need to be reemphasized.

Management must continue to support the methodology. One way that they can do this is to work change through the manual, not around it. In the case study, data processing management issued some directives independently of the methodology which were contradictory to the manual's precepts. As a result, the manual's credibility and authority were diluted. To maintain the effectiveness of your methodology, use it as the instrument to introduce changes in development practice.

The tail must not be permitted to wag the dog. The purpose of a development framework is to facilitate the design and successful implementation of good systems, not to become infatuated with counter-productive paperwork. Therefore, make your methodology

flexible. Some commercial products have offered separate methodologies complete with a variety of forms for use on projects of differing size. Your own approach need not go that far. Katch's advice to in-house developers relative to inflexibility is worth repeating:

"If the structure is inflexible, is procedural in nature, and adheres to a 'cook-book' style where one operates by the numbers, not only do unnecessary tasks have to be performed thus decreasing the staff's productivity, but the staff's creative powers are completely stifled."
(21)

Installing a methodology is an expensive proposition, whether it is a commercial or in-house product. This was demonstrated by the package prices quoted in Section 4.6 and the costs of the case study's in-house effort. Nevertheless, as development costs for new data processing systems are escalating because of inflation, installing a methodology now should prove to be a sound investment. This is particularly true if an objective of the framework is to produce systems which contribute to the firm's profitability.

Your Methodology Manual does not have to be the size of an encyclopedia. A small organization lacking the resources of the company in the case study can reduce the required effort in several ways. It may choose to 1) limit the number of included components and/or 2) reduce the amount of verbiage by referencing guidelines in various sources and textbooks, providing that the latter are readily available to all the staff. If you choose to restrict the number of components, make certain that none of the following are omitted:

- Project Acceptance and Approval Procedures
- Creeping Commitment Policy
- User Responsibility Policy
- Development Life Cycle including Phase Tasks
- Milestones and Milestone Documents
- System Assurance Reviews

The example of the case study is only one of many where a company has developed its own methodology. This was also indicated in the responses to the mail survey. The experience of the case company and those in the survey are proof that in-house development can be

accomplished and its product utilized with satisfactory results. These guidelines should enable systems and programming managers, should they attempt to follow its direction, to enjoy the fruits of information systems successfully developed through their own development methodology.

7.2 Areas for Further Study

According to the mail survey, considerable time and effort were required on the part of purchasers to customize and to implement their respective commercial methodologies (See Chapter One). It is not clear why commercial packages need such modification and so much user involvement, particularly when on-site vendor assistance is sometimes advertised as part of a sales agreement. An independent study of user experience could establish 1) the extent to which modification is necessary, 2) why customizing is needed, and 3) to what extent user resources are required.

Another area for study concerns the relationship between specifications and purchases of commercial methodologies. The author hypothesizes that most con-

tracts for the latter are negotiated prior to the development of specifications. The reason for this hypothesis is that it is supposed that many firms do not know what they need or want, and, without the benefit of guidelines such as this, would rather pay to find out. Others, as in the case study, may be reluctant to invest in a commercial package having found that, as a result of developing a specifications document, they need very little in the way of new methodology components beyond those they already have. A study of organizations with methodologies, either purchased or in-house products, could substantiate or refute this hypothesis.

The mail survey indicated that some firms were not totally satisfied with their methodologies. Further research, therefore, is needed to establish the reasons why they are considered less than satisfactory. The objective of this investigation would be:

- 1) to establish the criteria for adjudicating success or failure of a methodology,
- 2) to expose the factors contributing to failure, such as missing components or mismanagement, and

3) to document pitfalls which should be avoided
lest your methodology suffer a like fate.

BIBLIOGRAPHY

- 1) Nelson, B. S., "Who Needs Methodology?", ICP Interface, Data Processing Management; Winter 1977, p. 5-8, 19.
- 2) Yourdon, E., Design of On-line Computer Systems, (Prentice-Hall, Inc.); 1973, p. 95-123.
- 3) Metzger, P. W., Managing a Programming Project, (Prentice-Hall, Inc.); 1973, p. 13-14.
- 4) Toellner, J. D., "User Involvement and Enforceability - What, When, and How?", ICP Interface, Administrative & Accounting; Winter 1977, p. 7-13.
- 5) Glover, S. O., "Letters to the Editor", Datamation; May 1978, p. 25.
- 6) Frazier et al., "Developing a Project Management Package", Journal of Systems Management; December 1976, p. 39-40.
- 7) Katch, D., "The Makings of MIS - Part Two", Infosystems; July 1978, p. 56.
- 8) Anderson, H. V., "Plan for Process Success", Journal of Systems Management; December 1974, p. 14.
- 9) Epner, S. A., "Minimizing Epner's Law: The OOPS Factor", Infosystems; January 1978, p. 72.
- 10) Auerbach, "User-Oriented Systems Analysis and Design", Systems Development Management Series (Auerbach Publishers); 1976, p. 8-9.
- 11) Anthony, R. N., Management Accounting Principles, (R. D. Irwin, Inc.), 1970 revised edition, p. 456.
- 12) Daly, E. B., "Management of Software Development", Transactions on Software Engineering, (IEEE Computer Society) May 1978, p. 232-242.

- 13) Brooks, F. P. Jr., "The Mythical Man-month", Datamation; December 1974, p. 47.
- 14) Canning, R. G., "Project Management Systems", EDP Analyzer; September 1976, p. 6.
- 15) Johnson, R., "Developing Management to Develop Systems", Infosystems; August 1978, p. 87.
- 16) Hartman, W., et al., Management Information Systems Handbook, (N. Y.: McGraw-Hill); 1968.
- 17) Yourdon, E. and L. Constantine, Structured Design, (Yourdon Press); 1978 Second Edition.
- 18) Burch, J. G., Jr. and F. Strater, Jr., Information Systems: Theory & Practice, (Hamilton Publishing Co.); 1974.
- 19) Gildersleeve, T. R., Dataprocessing Project Management, (N. Y.: Van Nostrand Reinhold); 1974.
- 20) Unknown, 8-Stage Approach to Systems Development, GTE Data Services, Inc., (Diebold Research Program).
- 21) Katch, D., "Systems Development Methodologies - A Tutorial", ICP Interface, Data Processing Management; Winter 1977, p. 11-19.
- 22) Cohen, L. T., Data Base Management Systems, (Q.E.D. Information Sciences, Inc.); 1973.
- 23) McFarlan, F. W., "Management Audit of the EDP Department", Harvard Business Review, (Harvard University Press); May-June 1973, p. 140-141.
- 24) Miller, F. W., "Measuring MIS Performance Still a Problem", Infosystems; March 1978, p. 26.
- 25) Bryce, T., "Evaluating a Systems Design Methodology - Part II", Infosystems, January 1979, p. 60.
- 26) Dimmick, L. R., et al., Systems/Project Development Methodology, (Bethlehem Steel Corporation); 1978.

- 27) Hooper, W. G., et al., MIS Systems Methodology, (Monsanto Co.); 1976.
- 28) Canning, R. C., "The Production of Better Software", EDP Analyzer; February 1979, Vol. 17, No. 2, p. 1-13.

Exhibit A

QUESTIONNAIRE

1. How many persons are employed by your company?
1-100__ 101-1000__ 1001-5000__ Over 5000__
2. Indicate the largest computer system used by your company. _____
3. How many locations (sites, plants, divisions, etc.) are served by their own data processing organization? _____
4. How many Application Development (Systems & Programming) personnel are employed at all locations?

5. What types of applications are processed in-house?
On-line__ Batch__ Remote__ Time-sharing__
Message Switching__
6. Do you utilize a Data Base Management System (software package)? _____
7. Indicate a relative percentage of usage of the following programming languages.
COBOL__ RPG__ BASIC__ PLI__ FORTRAN__ BAL__
Other__
8. Does your data processing organization have a long-range planning procedure? _____
9. Does your data processing organization adhere to a formal "Project/System Development Methodology"?

10. If your answer to question 9 is "yes", do all locations participate? _____

Answer the following only if your answer to question 9 is "yes".

11. Does your "methodology" include, or advocate, or define a standard for any of the following?

Yes No

A "Project Life Cycle," i.e., development phases?

Definition of responsibilities, by phase?

Definition of tasks, by phase?

Project management techniques, e.g., PERT/CPM, Bar Charting?

Interfaces with L. R. Planning procedures?

Project acceptance/approval procedure?

Phase milestones?

Milestone documents, e.g., reports, proposals, etc.

Milestone document approval procedures?

"Good" development practices?

System Assurance or Quality Assurance Reviews?

Estimating technique?

Cost/Benefit Analysis procedures?

Risk Analysis procedure?

Hardware/Software selection guidelines?

Data base design methodology?

Program design methodology, e.g., Structured Design?

Post implementation review procedure?

12. Did you purchase your "methodology"? _____
13. If you purchased your "methodology", did you make any modifications to suit your shop? _____
14. How much time and effort did you spend developing, modifying, and implementing your methodology?

15. What was the elapsed time for completion of this effort? _____
16. If you developed your own methodology, please indicate who was directly responsible for its development (title & function)? _____
17. Was it a group effort? Indicate how many people wrote it. _____
18. Did you have any outside assistance (vendor, consultant)? _____
19. Did you have any guidelines to assist you in this effort? ____ If yes, describe briefly. _____

20. Has your methodology been successful? _____
21. Has your methodology had the support of top management? _____
22. Does anyone have a continuing responsibility for maintaining the methodology? _____
23. If you had it to do all over again, would you choose the same approach? ____ If no, please explain briefly.

Comments:

Name & Title: _____

Company: _____

**RETURN TO: W. A. NORRIS, JR., 3460 FOX DRIVE,
BETHLEHEM, PA. 18017**

Exhibit B

RESPONSES TO THE QUESTIONNAIRE

1. Number of employees in respondent's company:

<u>Number</u>	<u>With Methodology</u>	<u>No Methodology</u>
1 - 100	-	1
101 - 1000	2	8
1001 - 5000	5	6
over 5000	18	-

2. Largest computer system used:

IBM 3033	1	NCR 251	1
IBM MP 168	3	UNIVAC 9030	1
IBM 370/168 or 165	5	ITEL AS/5	1
IBM 370/158 or 155	11	IBM 360/40	1
AMDAHL 470/V6	1	IBM 360/22	1
CDC 6600/7600	2	IBM SYS/3	1
Burroughs B4800	1	Unanswered	3
DEC 2050	1		
IBM 370/135,138,148	6		

3. Number of sites with a separate data processing organization:

One	20
Two	1
Three	2
More than 3	15
Unknown	2

4. Number of systems development personnel at all locations:

	<u>With Methodology</u>	<u>Without Methodology</u>
1 - 10	2	9
11 - 25	3	5
26 - 50	4	1
51 - 100	4	-
over 100	12	-

5. Types of processing:

On-line	32
Batch	38
Remote	26
Time-sharing	15
Message Switching	6
Did not answer	2

6. Users of a Data Base Management System: 22

7. Program Languages used:

COBOL	35
RPG	20
BASIC	6
PL1	8
FORTTRAN	23
BAL	22
Other	8
Did not answer	1

8. Respondents with long-range planning procedures:

Yes	<u>29</u>	No	<u>11</u>
-----	-----------	----	-----------

9. Respondents adhering to a formal development methodology:

Yes	<u>25</u>	No	<u>15</u>
-----	-----------	----	-----------

10. Of those with methodologies, do all locations participate?

Yes	<u>14</u>	No	<u>10</u>	Unanswered	<u>1</u>
-----	-----------	----	-----------	------------	----------

11. Are these components included in your methodology?

	<u>Yes</u>	<u>No</u>
a. Project Life Cycle, i.e., developed phases	23	2

	<u>Yes</u>	<u>No</u>
b. Definition of responsibilities, by phase	22	3
c. Definition of tasks, by phase	22	3
d. Project management techniques, e.g., PERT/CPM, etc.	21	4
e. Interfaces with long-range planning procedures	13	12
f. Project acceptance/approval procedure	24	1
g. Phase milestones	19	6
h. Milestone documents, e.g., reports, etc.	20	5
i. Milestone document approval procedures	17	8
j. Guidelines, i.e., "good" development practices	21	4
k. System assurance reviews	16	9
l. Estimating technique	17	8
m. Cost/Benefit Analysis procedure	21	4
n. Risk Analysis procedure	4	21
o. Hardware/software selection guidelines	16	9
p. Data base design methodology	18	7
q. Design methodology, e.g., Structured Design	21	4
r. Post implementation review procedure	15	10

12. Methodology development statistics:

	<u>In-house</u>	<u>Purchased</u>
Number	20	5*
No. requiring modification	N/A	5
Avg. No. of writers	4.4	2.5
Avg. duration (yrs.)	3.0+	2.2
Avg. effort (man-years)	2.8	2.8

*One firm reported a cost of \$90,000 and another, \$50,000.

16. If developed in-house, who was responsible?

Manager, Systems Development	9
Parent company	1
Project managers	1
Sr. Staff Analyst	2
Programming (?)	1
Unanswered	6

19. Respondents claiming guidelines for developing their methodologies and their sources:

IBM Business System Structure	1
IBM PMS	1
GUIDE, SHARE, etc.	3
Guidelines by corporate	1
Experience at prior companies	2
Seminars	1
Standards developed by a consulting firm for another organization	1

20. Has your methodology been successful?

Yes	16
No	1
Not completely	4
Too soon to tell	3
Yes, after complete rewrite	1

21. Has your methodology had the support of top management?

Yes	19
No	4
Moderate	1
Other	1

22. Does anyone have a continuing responsibility for maintaining the methodology?

Yes	19
No	1
Management	4
Other	1

23. Number choosing the same approach, if task were to be done again:

Yes	19
No	4
Probably	2

VITA

Warren Arthur Norris, Jr. was born October 6, 1929 in Lebanon, Pennsylvania. His parents were the late Annetta Barron Norris and Warren Arthur Norris. After graduating from John Harris High School, Harrisburg, Pa., in 1947, Mr. Norris attended Yale University (New Haven, Connecticut), majoring in Industrial Administration (General Engineering). Upon graduation in 1951 with a Bachelor of Science degree, he entered the employ of Bethlehem Steel Corporation as a member of their management training program. Starting in the Manufacturing Division of the Bethlehem (Pa.) Plant, he advanced through various positions to Division Engineer in 1957. He became a Systems Analyst in the Bethlehem Plant Systems Department, when that department was created in 1959. Later, as Plant/Systems Coordinator on the staff of the General Manager of Corporate Data Processing, he was to be the writer of Bethlehem Steel's System/Project Development Methodology Manual, an effort which provided the case study for this thesis. He is currently assigned to the Accounting Department as a

Senior Analyst.

**Mr. Norris resides in Bethlehem, Pa. with his wife
and three children.**